PROGRAM FOR MOBILE PHONE PALMFORCE PERSONAL ORGANIZER

CHUNG FOO SHYN

FACULTY OF COMPUTER SCIENCE &
INFORMATION TECHNOLOGY
UNIVERSITY MALAYA
KUALA LUMPUR
2003/2004

University of Malaya

PalmForce

(Personal Organizer)

Student Name: Chung Foo Shyn

Student ID No.: WEK010062

Course Code: WXES 3182

Supervisor Name: Assoc. Professor Raja Nor Ainon

Moderator Name: Ms. Rafidah Mohd. Noor

Final Year Project

This project is submitted to the Faculty of Computer Science and Information Technology, University if Malaya in partial fulfillment of the requirement for the Degree Bachelor of Computer Science

Abstract

PalmForce is a Personal Digital Assistant (PDA) organizing application that allows the users to manage their daily activity and keep important data that they need. PalmForce is the main product to be developed in this project, which is one of the thesis titles proposed this year in Faculty of Computer Science and Information Technology, University Malaya. Our group, which consists of 4 students, will be required to work on different modules on this application, and integrate these modules later on. All the final year degree students in this institution are required to finish their final year project to graduate from the university.

PalmForce is divided into 4 modules, which are the personal organizer, financial organizer, educational organizer, and health organizer. Each of us will develop the different modules and integrate the product at the end stage, as mentioned earlier. After discussing within ourselves, we divided the modules to each of us. I will be working on the Personal Organizing tools modules, thus this documentation would consist mainly the information about the personal organizer. In Chapter 1 the objectives and overview of the Personal Organizer are being discussed. The overall function that is provided by this application is presented, together with the project scope as well. To develop this project properly, a lot of information and reference are needed. Thus research had been carried out in several aspect within this field, and the summary of the research can be found in Chapter 2.

To finish this project on time and learn how is the actual software development process goes on, a methodology is chosen as a guide throughout the project. The methodology chosen is the prototyping methodology. A prototype is a mock of the desired system that excludes some of the major function of the system. It is build to allow user to give their opinion on the product, thus allow the developers to change their product according to the users' needs. The introduction of this development module and the reason why it is chosen can be found on Chapter 3.

The functional requirements and the environment chosen to build this system are being discussed in Chapter 4. The result from the survey that is carried out is summarized here, and with this information, the functional requirements and non-functional requirements are determined. Chapter 5 gives a brief introduction to the basic architecture of the system. The data flow diagram and some screenshot from the prototype are presented here.

Chapter 6 discussed briefly the implementation of this module. This chapter gives description about how the tools and developing environment are set up before the real coding take place. It also gives the method used to do the coding. Chapter 7 is a summary of the testing done on the system. It contains the method used to test the system and the result of these tests. Finally Chapter 8, which is system evaluation, gives the information about how evaluation is done on the system. The information included here are the strength of the system, the weak point and problem face, and future enhancement for the system.

Acknowledgement

It had been a great experience for me while I was working hard to finish this first part of my final year project. With the help from the people around me, I manage to finish it on time and proceed to the second phase. I would like to thank these people, whose help has been critical while I'm working on my task.

First of all, I would like to thank my supervisor for this project, Prof Madya Raja Nor Ainon Raja Zainal Abidin. Her constructive advice, generous guidance, encouragement and full support on our group enable us to work on our job without much hassle. And most of all, she allow us to propose our own idea, which is the core concept of this project. Her enthusiasm is mostly appreciated.

Another lecturer to thank to is Puan Rafidah Mohd. Noor, who is the moderator for this project. Her contributions of suggestion and ideas allow me to understand the real requirements and constraints for this project. Special thanks to all my project partners as well, who are Mr Leong Yoon Fui, Mr Loh Chee Wai and Mr Ng Heang Tung. They had help me a lot to finish this project. We discussed the whole project and worked together by sharing ideas and support within these few months.

Finally special thanks to all those people who dedicate their precious time to answer our questionnaire and allow us to interview them.

Table of Content

List of Figures

List of Table

Chapter 1 Introduction

1.1 Project Overview

Personal Digital Assistant (PDA) is gaining popularity nowadays as a managing tools for a wide range of users, be it professional workers or businessmen. PDA is easy to carry, and its processing power and space is getting more powerful with each new release. The introduction of this gadget has change the way people organize their information. Now PDA users can keep their important information on the PDA and refer to them easily.

This project is carried out to develop an organizer that would aid the PDA user to manage their daily activities and their data. The main system is divided into 4 modules, which are the personal organizer, financial organizer, health organizer and educational organizer. Each module has its on unique features that would allow the user to store relevant information and manage them easily.

Traditional personal organizer allow user to record their personal details, such as name, address, date of birth and so on. So does the personal organizer module in this project. It provides a platform where user can set the reminder to remind them of certain event that they selected. It also features a contact list where user can input the details of their contact in the system and refer to them whenever they want. This module contain an alarm as well, that is the user would be able to set the PDA as the alarm to wake them up,

or remind them of certain time. In short, this module would cover the aspect that most of the digital organizer would have today, which is:

- as a reminder for the user. User can store the activities that need to be done in the PDA and set it to alarm the user when the time comes. Calendar is available for user to use too. User then can plan their daily schedule easily.
- as a contacts storage. Address book is a very useful tool for those that has a lot of connections with people, especially businessman. Address book in the PDAs not only store the data but also arrange them in index so that the user can browse through them easily, even if the list contains over 100 names.
- as a small accessory tool. Organizer comes with some small useful tools for users to use in their daily life too, such as alarms, calculator etc.

Basically this module would be useful to any user from any range of age. The project is proposed to be developed using Java, which would allow it to be install in different operating system in the PDA. This would allow the user to use this program in different brand of PDA. Furthermore, the additional modules that comes together would make it easy for the user to install just one package of programs in the PDA.

1.2 Project Objectives

This module is developed base on the following objectives:

1) *To provide a set of organizing tools for PDA users*

   This module would provide the most basic organizing tools for the user, such as the to-do list and the contact list to allow user to save important information that they need in their PDA.

2) *To practice paperless environment*

   By using this module on a PDA, user would not need to use pen and paper to write down their information or notes. This could help to encourage the use of paperless environment.

3) *To provide an easy way to manage personal information*

   This module would allow its user to manage their personal information more systematically and store them in a save place. By using this module on a PDA user would not need to worry that their important information would be left in unknown places or the information would be damaged as it's written in a paper.

4) *To serve as a convenient reminder tool for the user*

   Reminder is a very important features that many organizing tools offers nowadays. PDA user would need this function to alert themselves of certain event that they require attention. This module includes this function as well.

1.3 Scope and Limitation

As stated in the introduction, this module is mainly targeted to all type of users. Such coverage means that it would include functions that can be use by everyone. But still there would be limitation in providing this module. Below are the scope of this module and the limitation that it carries.

1.3.1 Scope of the module:

- Stores the general personal information about the users
- Let the user to store and browse through contacts data, including name, phone number and address.
- Provide workspace for user to schedule their activities daily in a to-do list.
- The user can set reminder on occasion that they picked base on the time and date of the occasion.

1.3.2 Limitation:

- Limited processing power. Therefore the data is being process at a slower rate.
- Sorting is difficult. Because that the database contain quite a number of data, it will slow down the whole module if sorting is done. Therefore it is omitted.
- Limited user control on the interface.
- Synchronization with PC hard to done.

1.4 Expected Outcome

The expected outcome for this module is as followed:

- A module that would allow the user to keep and manage their personal information.
- A module that provide the alert user function, which can be used as the reminder tool by the user.
- A module that can be integrated with other module to produce a complete organizing system for PDA user.
- A module where enhancement and upgrades can be done so that future modification can be done due to the user requirements.

1.5 Project Schedule

A schedule that contains the different development stages for this project is being planned out. The schedule is very important as it serves as the guidance and time-management tool during the development process. It helps to ensure that the module can be developed on time as well.

Below is the Gantt chart for this project:

ID	Task Name	Duration
1	Preliminary Study and Planning	14 days
2	Literature Study	25 days
3	System Analysis	35 days
4	System Design	30 days
5	Prototyping	14 days
6	Development and Coding	84 days
7	Unit Testing	30 days
8	System Testing	25 days
9	Implementation and Maintenance	20 days
10	Documentation	217 days

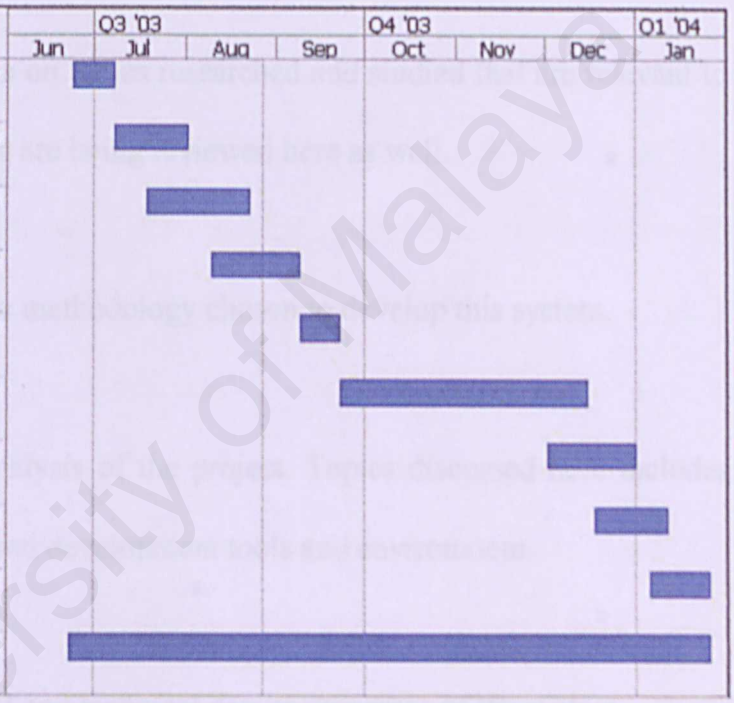


Figure 1.1: Project Schedule

1.5 Report Summary

This project's documentation is divided into several chapters. For the first report (WXES3181) the first 5 chapters will be included. The rest will be included in the second report (WXES 3182). The major layouts of the chapters are:

Chapter 1: Introduction

This chapter gives as overview of the project, includes the project objectives and scopes.

Chapter 2: Literature Review

This chapter gives brief explanation on topics researched and studied that are relevant to the project. Similar existing systems are being reviewed here as well.

Chapter 3: Methodology

This chapter reveals the detail of the methodology chosen to develop this system.

Chapter 4: System Analysis

This chapter emphasized on the analysis of the project. Topics discussed here includes the project's requirements, the desired development tools and environment.

Chapter 5: System Design

This chapter explains the conceptual and technical design processes of the system.

Chapter 6: System Implementation

This chapter explain how the system will be implemented into a real PDA device.

Chapter 7: Testing

This chapter reveal the types of testing that are done to ensure the end product will performed as it is built to be.

Chapter 8: System Evaluation

This chapter explains the process to evaluate the end product.

1.6 Chapter Summary

This chapter describes the module to be developed. The objectives of the project is being listed at the earlier part of the chapter, including to develop a module that would provide the basic organizing function to the PDA users. Beside that, the scope of the project is provided in this chapter as well. The project's scope is the coverage of the project. It describe what is done by the system.

However, the project also does carries some of the limitations that will effect the development process of the project. An overview of this limitation is provided here. The limitation is being listed out so that it can give a clear description about the module to be developed to the developers. Expected outcomes is described as well as it can provide a guideline in order to develop a module that will meet the users' requirements.

The project schedule is included in this chapter too. It is a time arrangement towards the completion of the project. This chapter also stated the summary of the content for each chapter that is included in this report.

Chapter 2 Literature Review

2.1 Introduction

In the process of developing an existing or new system, lots of activities need to be carried out to gain relevant information on the project. Researches become the most important and major activity before the development phases start. Research ought to be carried out in order to study the problem or project constraint thoroughly. It should be performed until a complete understanding of the system is gained. Another important purpose of a literature review is to sufficiently equip the developer with some knowledge of the strengths and limitation of several development tools. This can help the developer to choose the right tool in system development.

The studies in this project have reviewed the technology in developing programs for mobile devices. It also present some of the existing system that have the similar function as this project.

2.2 Programming Languages

A programming language is an artificial language for expressing computer programs. It is an artificial language consisting of a vocabulary along with grammatical rules used to write a set of instructions that can be translated into machine language and then executed by a computer. Machine language is the language that computer actually understands. Each different type of CPU has its own unique machine language.

There are a few numbers of programming languages that support mobile programming nowadays. For example, C++, Visual Basic, and Java can be used as the development programming language to develop a mobile program. Below is some short description on the programming languages.

2.2.1 C++

In the early 1980, Bjarne Stroustrup at AT&T Bell Laboratories designed C++ as an extension to the C language, providing data abstraction and object-oriented programming facilities. Stroustrup designed the language to remain compatible and comparable with C in terms of syntax, performance and portability. The C++ language provides key capabilities and benefits offered by object-oriented programming. C++ excludes features that would constrain its use to a limited set of application domains and environments. The mechanisms are defined to allow highly efficient implementations and versatility offered by the language.

Some of the C++'s features and advantages are listed below:

- ***Object-oriented programming***

  The possibility to orientate programming to objects allows the programmer to design applications from a point of view more like a communication between objects that on a structured sequence of code. In addition it allows the reusability of code in a more logical and productive way.

- ***Portability***

  The same C++ code can be compiled in almost any type of computer and operating system without making changes. It is obvious that C++ is one of the most used and ported to different platforms programming language.

- ***Brevity***

  Code written in C++ is very short in comparison with other languages, since the use of special characters is preferred before key words, saving effort (and prolonging the life of our keyboards).

- ***Modular programming***

  An application's body in C++ can be made up of several source code files that are compiled separately and then linked together. Saving time since it is not needed to recompile the complete application when making a single change but only the file that contains it. In addition, this characteristic allows to link C++ code with code produced in other languages like Assembler or C.

- ***Speed***

  The resulting code from a C++ compilation is very efficient, due indeed to its duality as high-level and low-level language and to the reduced size of the language itself.

2.2.2 Java

Java is a powerful computer programming language that is fun to use for novices while simultaneously being appropriate for experienced programmers building substantial information systems. Java is certain to become the language of choice in the new millennium for implementing Internet-based and Intranet-based applications and any other software for devices that communicate over a network.

Java is a full-featured computer language that incorporates the best of modern thinking about Object Oriented (OO) programming. Java is simpler and more robust than other computer languages and combines features, which make it ideal for programs, which must deal with networks. The designers of Java emphasized securities, ease of programming and independence from any particular hardware. These features brought Java near instant acclaim in the programming word and meteoric rice in public consciousness.

Java API can be divided to 3 platforms that each suit different kind of users. These 3 platforms are: Java 2 Enterprise Edition (J2EE), Java 2 Standard Edition (J2SE), and Java 2 Mobile Edition (J2ME). J2EE is the platform for developing large system where commercial demand is very high. J2SE is the standard platform where the most basic functions of Java can be found. J2ME is the platform where it is used to develop programs for mobile devices. Figure below shows each type of the platform.

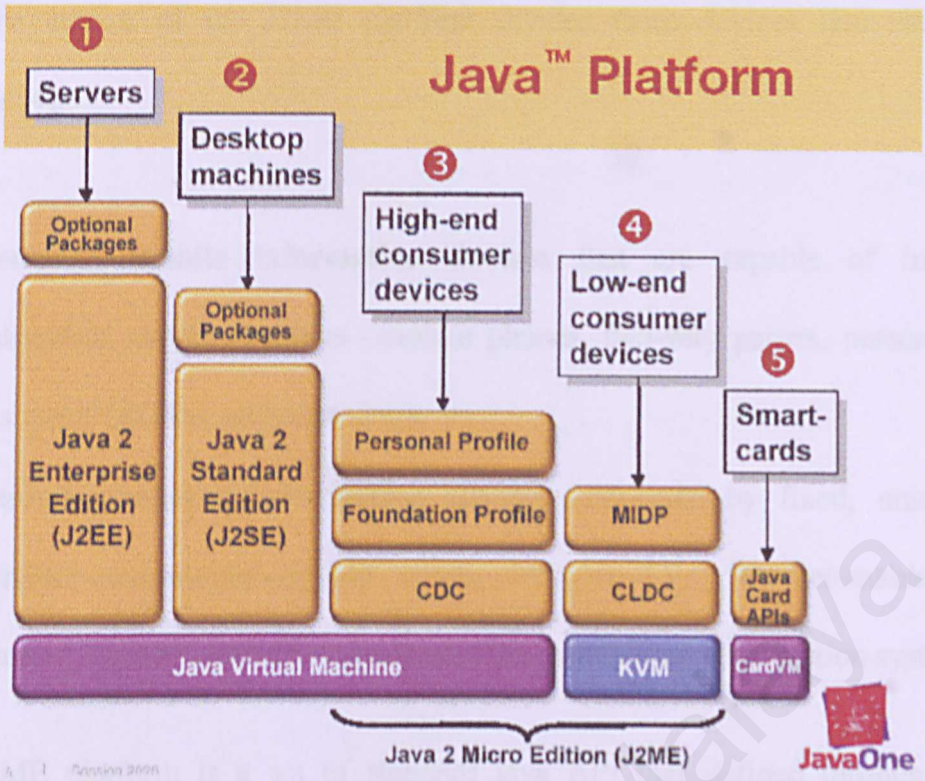


Figure 2.1: Java platform at a glance

Java 2 Mobile Edition (J2ME)

Java 2 Platform, Micro Edition was introduced to meet the demand of the information appliances in the rapidly developing consumer and embedded markets. The Java application development environment has enable developer to develop new system for many new and powerful information appliance products. Java technology also enable users, service providers and device manufactures to take advantage of a rich portfolio of application content that can be delivered to the user's device on demand, by wired or wireless connections.

The Java 2 Platform, Micro Edition (J2ME) is the Java 2 platform targeted at consumer electronics and embedded devices like wireless phones, pagers, personal digital assistants, camcorders, game devices, small retail payment terminals and smart

cards. The creator of the J2ME platform divides these devices into two distinct categories [4]:

- **Personal, mobile information devices** that are capable of intermittent networked communications—mobile phones, two-way pagers, personal digital assistants (PDAs), and organizers
- **Shared-connection information devices** connected by fixed, uninterrupted network connection—set-top boxes, Internet TVs, Internet-enabled screen phones, high-end communicators, and car entertainment/navigation systems

J2ME platform is a set of standard java API that defined through the Java Community Process program. Java Community Process included leading devices, software provider and service providers. It lets the devices almost as powerful as computer. J2ME platform offer developer a flexible user interface, robust security model, board range of built in network protocols and support for networked and disconnected application. By using J2ME, developer can "Write Once, Run Anywhere"

J2ME platform technology has three components which taken together to form a compliant Java application environment. A J2ME compliant Java application environment requires a configuration and a profile. Optional Packages provide device designers and other Java Community Process (JCP) participants with a standards-based extension framework [5].

- **Configuration** is a combination of a Java virtual machine and a set of application support APIs that are shared across a class of devices. It is a specification that defines the characteristics and the facilities available. It also specifies a minimum set of features for a category of devices.
- **Profile** is a set of APIs that address the needs of a narrower device category. The profile specifies the application-level interface for a particular class of devices. A profile implementation consist of a set of Java class libraries
- **Optional Package** is a set of technology-specific APIs that extends the capabilities of a Java application environment. RMI Optional Package and JDBC Optional Package are two Optional Packages.

A profile is implemented on the top of a configuration that is closer to the implementation or the real-world applications. Applications are then built on top of the configuration and profile and can use only the class libraries provided by these two lower-level specifications. Table below shows the different type of the J2ME profile.

Profile	Description
Mobile Information Device Profile (MIDP)	Designed for mobile phones and entry-level PDAs. It offers core application functionality required by mobile application such as user interface, network connectivity, local data storage and application management. Combined with Connected, Limited Device Configuration (CLDC), it provides a complete Java runtime environment that leverages the capabilities of handheld devices and minimizes both memory and power consumption.
Foundation Profile (FP)	FP is the lowest profiles for Connected Device Configuration (CDC). It provides a network-capable implementation of CDC that can be use for deeply embedded implementation without a user interface. It can combined with Personal Basis Profile and Personal

	Profile for devices that require a graphical user interface (GUI)
Personal Profile (PP)	PP is the CDC profile aimed at devices that require full GUI or Internet applet support. It includes the full Java Abstract Windows Toolkit (AWT) libraries and offers Web fidelity, easy running Web-based applet designed for use in a desktop environment.
Personal Basic Profile (PBP)	PBP is a subset of PP provides an application environment for network connected devices that support a basic level of graphical presentation or require the use of specialized graphical toolkits for specific application.

Table 2.1: J2ME profile

Java had defined 2 sets of configurations to avoid conflicts and fragmented landscape of incompatible platforms. These two configurations represents 2 categories of connecting devices that currently exist:

- Connected Limited Device Configuration (CLDC) – support personal and temporarily connected devices.
- Connected Device Configuration (CDC) – support constantly connected devices.

The relationship between these 2 configurations is that the they are a subset for each others. Although they are independent of each other, they should not be used together to define a platform.

The Connected Limited Device Configuration (CLDC)

Device Configuration (CLDC supports personal, mobile devices, which constitute a significantly less powerful class of devices than the one that the CDC supports. The CLDC specification identifies devices in this category as having the following characteristics:

- 160 to 512 KB total memory available for the Java platform

- 16-bit or 32-bit processor
- low power consumption, often battery powered
- intermittent network connectivity (often wireless) with potentially limited bandwidth

CLDC specifies the level of support of the Java programming language required, the required functional support of a compliant Java VM, and the set of class libraries required. Because of the device-constrain, a number of features have been eliminated from a J2SE Java virtual machine supporting CLDC. Below are some of the limitation:

- **No floating-point support:** The CLDC does not support floating-point numbers. Therefore, no CLDC-based application can use floating-point numbers or types such as `float` or `double`. This is mainly because CLDC target d e vices do not have floating point support in their hardware.
- **No finalization:** The CLDC APIs do no include the `Object.finalize` method so one cannot perform final cleanup operations on object data b e fore the object is garbage collected.
- **Limited error handling:** Runtime errors are handled in an implementation specific manner. The CLDC defines only three error classes: `java.lang.Error`, `java.lang.OutOfMemoryError` and `java.lang.VirtualMachineError`. Non-runtime errors are handled in a device-dependent manner that involves terminating the application or resetting the device.
- **No Java Native Interface (JNI):** A Java virtual machine supporting the CLDC does not implement the Java Native Interface (JNI) primarily for security

reasons. Also, implementing JNI is considered expensive, given the memory constraints of CLDC target devices.

- **No user-defined class loaders:** A Java virtual machine supporting the CLDC must have a built-in class loader that cannot be overridden or replaced by the user. This is mainly for security reasons.
- **No support for reflection:** Because reflection is not supported, there is also no support for RMI or object serializations.
- **No thread groups or daemon threads:** While a Java virtual machine supporting the CLDC implements multithreading, it cannot support thread groups or daemon threads.
- **No weak references:** No application built on a Java virtual machine supporting the CLDC can acquire weak references.

The benefits of CLDC are:

- Cross-Platform - Work is transferred between CLDC and other devices.
- Dynamic Content - Content is determined by user experience, and information transfer between CLDC and other devices.
- Security
- Developer Community - The developer talent needed for these devices already exists and is readily available for CLDC devices.

The Connected Device Configuration (CDC)

CDC is designed for devices that have more memory, faster processor and greater network bandwidth, such as TV set-top boxes, residential gateways, in-vehicle

telematics systems and high-end PDAs. CDC was designed around the two goals of J2SE compatibility and support for resource-constrains devices. CDC includes a full-featured Java virtual machine called the Compact Virtual Machine (CVM) and a much larger subset of the J2ME platform that CLDC. It also support floating point and core library features such as full class loading, thread support and security. Most of the CDC-targeted devices have 32-bit CPUs and a minimum of 2MB of memory available for the Java platform and associated application.

The benefits of CDC are:

- Enterprise usage – provide data integration with enterprise application and good security features for the mobile devices.
- Users – provides secure measure for end user, and solve the predictable user-experience.
- Device vendors – provide engagement with enterprise applications and secure the mobile application platform.
- Developers – provide rich API for programming usage, enhance safety and productivity of the Java programming language, target at multiple devices and leverage J2SE skills.
- Service providers – provide scalable offering to enterprises, and secure the mobile applications.

Virtual Machines (VM)

As known to all Java application developers, Java programs requires virtual machines (VM) to be installed on the platform before these programs can be compiled and executed. For CLDC and CDC there is no exception. The virtual machine available for

CLDC is Kilobyte Virtual Machine (KVM), and for CDC is the Compact Virtual Machine (CVM). The following are the description for these virtual machines:

Compact Virtual Machine (CVM)

Although CVM supports the same features as the J2SE VM, it is designed for consumer and embedded devices. So the CVM was reengineered to suit the constraints of limited-resource devices. The features of the CVM are:

- advanced memory system
- small average garbage collection pause times
- full separation of VM from memory system
- modularized garbage collectors
- generational garbage collectors
- generational garbage collection
- fast synchronization
- small class footprint
- mapping Java threads directly to native threads
- support Java 2 VM features and libraries such as security, weak references, Java Native Interface (JNI), Remote Method Invocation (RMI), Java Virtual Machine Debugging Interface (JVMDI)

Kilobyte Virtual Machine (KVM)

The VM that comes with the CLDC reference implementation is called the Kilobyte Virtual Machine (KVM), so named because it uses only a few KB of runtime memory. The KVM is a compact, portable Java virtual machine specifically designed from the ground up for small, resource-constrained devices. The high-level design goal

for the KVM was to create the smallest possible "complete" Java virtual machine that would maintain all the central aspects of the Java programming language, but would run in a resource-constrained device with only a few hundred kilobytes total memory budget [10].

KVM was designed to overcome three key technical challenges:

i. reducing the size of the virtual machine and class libraries themselves

ii. reducing the memory utilized by the virtual machine during execution

iii. Allowing for components of the virtual machine to be configured to suit particular devices (for example, by allowing pluggable garbage collection)

JVM is designed for:

1) **Reduced VM Size** - the K virtual machine is currently only 50-80 K of object code in its standard configuration, depending on target platform and compilation options.

2) **Reduced Memory utilization** - In addition to the K virtual machine's small object size, it requires only a few tens of kilobytes of dynamic memory to run effectively. Because of the reduced VM size and memory utilization, even with total memory available of only 128K the K virtual machine enables useful Java technology-based applications to run on a device.

3) **Performance** - the K virtual machine is able to run effectively on 16 bit processors clocked as low as 25 MHz, and can scale smoothly up to much more powerful 32 bit processors.

4) **Portability** - although implemented in native code for extra performance, the K virtual machine has a highly portable architecture that reduces system dependencies to a minimum. Even multi-threading and garbage collection have

been implemented in a completely system-independent manner, enabling speedy porting to any host platform.

2.2.3 Comparison between C++ and Java

Criteria	Java	C++
API reference	Uses MIDP API as reference. Limited functions over the operating system but easier to learn. Comes in high-level language nature.	Uses the Palm OS API as reference. More controls provided but hard to learn. Comes in low-level language nature.
Error handling	Does not need reset when the resources on the device are exhausted. User can regain control on the operating system.	Requires hard-reset to reset the system when the resources on the device are exhausted.
Graphical User Interface	Provide classes to build user interface. Interface is build faster with moderate quality.	A lot of time is spent to build base classes for interface. Interface is build slower but with good quality.
Floating point math	Not supported	Supported
CPU	Uses more CPU time	Uses less CPU time
Memory usage	Limited to 64K. Comes with a garbage collector.	Not limited to 64K. Better utilize of the space available.
Multi-threading	Supported	Not Supported
Virtual Machine	Requires a virtual machine to run.	Does not require virtual machine to run.

Table 2.2: Comparison between C++ and Java

2.3 Operating System

Software are designed to control the hardware of a specific computer system in order to allow users and application programs to employ it easily. The operating system mediates between hardware and application programs. It handles the details of sending instructions to the hardware and allocating system resources in cast of conflicts,

Operating system on a PDA is different from the operating system on a PC. Because of the hardware constraints with the PDA, such as the memory size, PDA's operating systems are build in small capacity, simpler and work with specific functions that comes with the PDA, such as the touch screen functions. There are only a few operating systems for PDA in the market now, with the Palm OS and Windows CE dominating. Below are the brief descriptions on the operating systems.

2.3.1 PalmOS

Palm originally was a software firm, creating handwriting-recognition software for the handhelds of other companies. When they invented GRAFFITI (the same handwriting-recognition that is still used in today's Palm OS devices) in 1994, most of today's competitors just ignored the ease of use and the high recognition rate of the new system and did not want to use it – because it required some additional work from the user, she had to learn the special Graffiti Strokes. Palm then decided that they did want to bring their product to market. And because nobody would create a hardware device to fit that purpose they just created their own PDA3 and named it PalmPilot, fitted with Palm OS 1.0. Soon the device boomed, there was no real competition - for the reasons

thatstill count for today's success of the Palm OS devices. Fact is: The current market share of Palm OS handhelds lies between 45% and 80%, depending on who makes the questionnaire and how a handheld is defined. Nonetheless Palm did not always have an easy time, they were challenged by their competitors, namely WINDOWS CE and PSION'S EPOC. This challenge lead to red figures, and Palm had to react. The firm did so by splitting into two firms in 2001, each one responsible for one of the two core competences of the OS[4]. The firm responsible for the OS licensed it to several hardware providers, one of them being Palm itself . Other ones include HANDSPRING and SONY. Palm PDAs typically come with several basic productivity applications, such as: Address, Calculator, Date Book, Expense, Memo Pad, Note Pad, To Do List.

Designing applications for Palm powered PDA is rather different from developing application for a high-powered Pentium PC which users tend to sit down and work for long hours. Palm powered handhelds, on the other hand, are a low-power, small, battery-operated devices used frequently but intermittently. There are a few advantages that inherited in Palm OS that makes it the most popular handheld's operating system today. Below are some of them [1]:

1) **Easy to use**

The Palm OS platform's ease of use means low training costs, high productivity, and fast deployment. Typical usage sessions on a handheld are shorter than on a PC -- the user gets in, captures or accesses data, and gets out. In that model, a user can't afford to fumble around with nested dialog boxes and hidden commands. The system's design must anticipate user needs and take care of them in advance. The designers of the Palm OS think like a user, sweating the details

of how a product will actually be used and how to make that simple and lightning-fast.

2) **The widest range of software and hardware**

Palm Powered systems are customizable to the needs of an individual user or of a corporate department. User needs vary tremendously, and Palm OS's "one size fits one" philosophy matches their needs much better than the "one size fits all" design used by other handheld systems.

3) **Easy to carry**

The Palm OS is designed just for mobile devices. It uses memory and battery power very efficiently, enabling Palm Powered hardware companies to design extremely small and lightweight systems. Palm Powered handhelds start at less than four ounces and five cubic inches (that's smaller than most wallets), and their batteries can last up to weeks without a recharge.

PalmOS Architecture

In order to provide the basic characteristics of mobile computing to handheld devices, PalmOS has been designed very diligently into a layered, component architecture. From the start, the flexible, extensible PalmOS has been designed to grow and evolve in response to user needs. The open, modular architecture allows developers, licensees, alliance and OEM partners to develop innovative new products and applications for a rapidly expanding global user market. Following is the description of PalmOS architecture and its interaction with other system components.

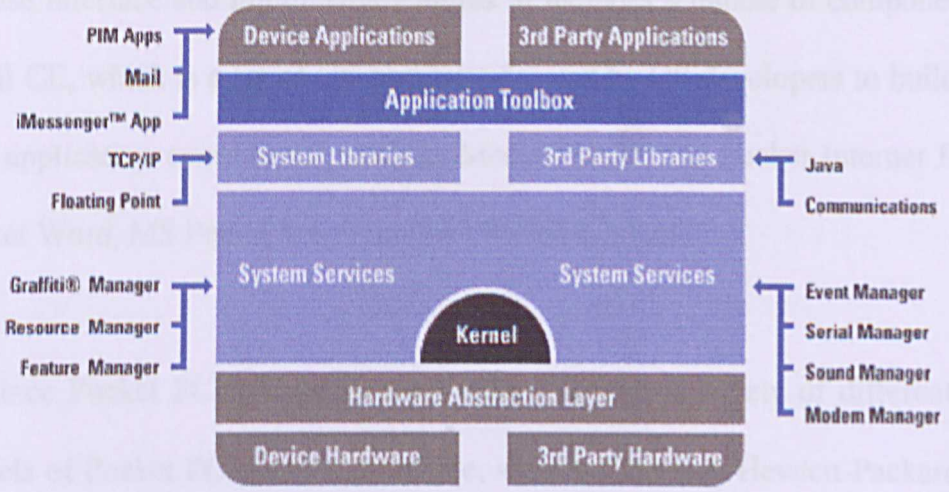


Figure 2.2: Architecture of PalmOS

The OS is built around the kernel, which handles all low-level communication with the processor, interrupts, provides multitasking and message queuing to the OS that resides on top of it. The kernel accesses the hardware via a hardware abstraction layer (HAL). On top of the kernel, there are the system services. Those are actually mapped ROM commands, so they are pretty fast in execution. Based upon the system services are the librairies, which are either produced by the OS section of Palm or by the individual licensees. On top of those librairies then are the applications, some of them with an individual or somehowstandard (per company and / or developer, this is very varying) toolbox.

2.3.2 Pocket PC

Pocket PC is another version of PDA that is getting more popular among the PDA users. Pocket PC is a software that includes operating system and useful application components in a package. Developed by Microsoft, it is aimed at providing

easy-to-use interface and multimedia features. It includes a bundle of components from Microsoft CE, which is a set of OS components used by OS developers to build custom OS, and application components, such as Media Player, MS Pocket Internet Explorer, MS Pocket Word, MS Pocket Excel and MS Pocket Outlook.

Since Pocket PC is manufactured independently, a variety of different makers and models of Pocket PC are now available, such as Compaq, Hewlett-Packard, Casio, Toshiba, NEC, Fujitsu, etc, but all Pocket PC have one thing in common, that is they are all powered by Microsoft Windows CE (Windows CE 3.0, in particular). This gives them the same level of compatibility and interoperatability that existed within desktop Windows. Better still, they can be connected to most Windows PC's to transfer, backup or synchronize the user's data.

Pocket PC 2002 is a revised edition of Pocket PC. It was code named "Merlin" and it replaces Pocket PC 2000 (code named "Rapier"). It is the latest Pocket PC interface to run on the Windows CE 3.0 platform. Pocket PC 2002 only supports ARM based processors, but comes with many new upgrades and features. With Pocket PC 2002 Phone Edition, the PDA itself can be used as a phone as well, bringing more convenience to the users. Below are some of the factors that make Pocket PC getting more popular today:

- **Familiarity to Windows-based system**: Pocket PC has the same interface as the desktop windows, thus users who had used Windows for years will find it easy to use.

- **Better Web browser**: The mini Windows Explorer in the Pocket PC is one of the best web browsers for a PDA today.
- **Easy multitasking**: Pocket PC allows users to open more than one application at a time, although only one application screen can be appear at a time. This makes it a bit faster to jump between applications.
- **MP3 player**: Pocket PC comes in standard with Windows Media Player for MP3 and other multimedia format file.
- **Wide range of support software**: With Pocket PC, supportive software can be found through web or vendors that suit the users need easily.

2.3.3 EPOC

EPOC is an operating system introduced by Psion, the first major manufacturer of PDA. It is an OS designed for small, portable computer-telephones with wireless access to phone and other information services. To earlier systems, EPOC adds wireless communication and architecture for adding application programs. Psion declared its first version of EPOC to be an open operating system and licensed it to other equipment makers. Psion then formed a new company with Ericsson, Nokia, and later Motorola called Symbian, which now licenses EPOC and continues to develop it. For portable equipment manufacturers, EPOC is an alternative to Microsoft's Windows CE. (3Com's PalmPilot uses its own proprietary operating system, PalmOS.) [2].

EPOC is a 32-bit, multi-tasking operating system that supports a pen-based graphical user interface (GUI). It is written in the C++ programming language using an object-oriented design. The code is very compact so that it can fit on a small ROM chip.

In addition to basic services, the operating system comes with an "application suite," that includes a word processor, e-mail handler, spreadsheet program, a scheduling application, general purpose database, sketch program, world clock, voice recorder, spell checker, calculator, communication programs, and a Web browser. EPOC can be scaled from relatively large configurations for a fully-functional handheld computer to small configurations for embedded applications.

Although EPOC can be ported to other microprocessors, Symbian's preferred platform is the Advanced RISC Machines (ARM) architecture. Symbian considers ARM the best platform in terms of millions of instructions per second (MIPS) per watt and per dollar cost. Symbian provides development kits for C++ and for OPL, a BASIC-like language. Java support is in beta test. Programmers write programs at a PC and use an emulator to test them. Several reasons why an EPOC is chosen by its users are:

- **It is a stand-alone miniature computer**: Most of the others PDA nowadays are designed to work closely with a PC. Whereas with EPOC, users can do all the things with it and independent of the PC.
- **It requires few memory spaces**: EPOC runs with small amount of memory, thus allowing others application to runs smoothly.
- **A lot of built-in office-type application**: EPOC comes with a lot of handy application that is useful to daily operations, without the hassle to install them manually.
- **Good multitasking features**: EPOC can work with multiple task with great handling. Users can run multiple application at the same time, each handling a different file while using EPOC.

2.3.4 Comparison between PalmOS, Pocket PC and EPOC

Criteria	Palm OS	Pocket PC	EPOC
CPU requirement	Does not need speedy processor to run	Requires high-end processor to run. Currently work with ARM processor only	Runs most mobile processor
Input method	Uses pen and Grafitti writing	Uses pen, handwriting recognition function are available too	Small thumb keyboard
Memory usage	Relatively small memory usage, and support limited external storage method	Needs a lot of memory to run. Provides a lot of internal memory, usually up to 32 – 64mb. Support additional memory as well	Needs small amount of memory to run. Usually provides up to16mb of internal memory. Support additional Flash memory
Data Back Up	Back up all data using HotSync as the conduit	Back up some basic data through ActiveSync. Additional steps are needed to back up extra data	No conduit is used. All data can be transfer to any platform except Machintos.
Multiple task	Does not support multi tasking	Support multi tasking but limited	Support multi tasking greatly and provide elaborate memory management
Multimedia and web application	Does not support multimedia files, can	Support media player and include internet explorer	Does not support multimedia format, include

	support moderate web browsing program		web browser but does not support frame

Table 2.3: Comparison between PalmOS, Pocket PC and EPOC

2.4 Review of existing similar system

PalmForce's Personal Organizer is a module that allows users to plan their daily schedule and set reminder for certain important events using the Palm devices. It is useful and provides easy-to-use interface for its users. In the market today, there are many similar products that provide more or less the same functions. Below are some of them and a table that introduce some of the basic features.

2.4.1 MyWeek

MyWeek is a pictorial, weekly planner that interfaces with the standard To Do List and Date Book applications. It runs in Palm OS 3.0 or higher and it fits in most of the Palm Devices available today. MyWeek allows you to:

- Drag To Do & Date Book items from one day to another or to another week
- Toggle between split and full Sat/Sun views
- Organize your To Do List from within MyWeek
- Organize your Date Book from within MyWeek
- Expand each day to a full screen view
- Jump directly to items in the To Do List & Date Book
- Jump directly to the Date Book page for any day of the week

- Color-code To Do & Date Book items by category

MyWeek utilizes established preferences for To Do List sorting order, "Show Completed Items," "Record Completion Date," Date Book Start and End Times, Alarm Preset, and overall handheld privacy settings. This application includes support for:

- International users
- Palm Tungsten 320 x 320 high-resolution display
- Sony Clie 320 x 320 high-resolution display
- Sony Clie 320 x 480 high-resolution display and virtual silkscreen
- Handera 330 high-resolution display and minimized graffiti area

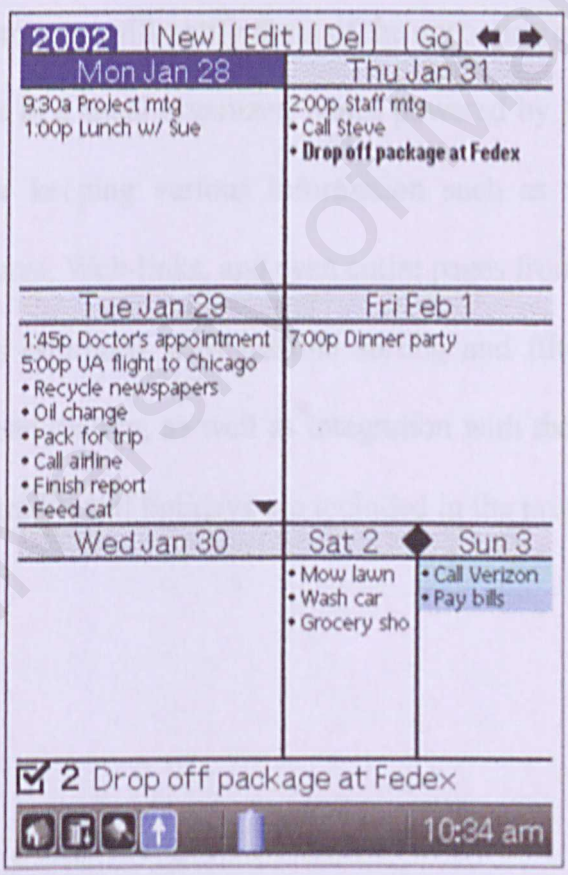


Figure 2.3: MyWeek

2.4.2 WinOrganizer

WinOrganizer is a full-scale personal information manager (PIM) that helps its users to organize and plan their business and personal life. It's a planner, notepad and address book combined in one powerful application.The organizer comprises all features of its paper prototype in a more comfortable way, which makes working with it quicker and easier. Its user-friendly interface combines all the users' notes, bookmarks, holidays, reminders, appointments, task lists, contacts, and passwords into a coherent tree outline form that can be customized as required. The data can be protected by password to limit the access to the users' personal information. Users will be able to track everyday tasks and, what's more, the program will notify them of the upcoming events and tasks.

In addition, like in a usual organizer, Notes powered by MS Word-like rich-text tools perfectly suit for keeping various information such as notes, recipes, famous quotations, tables, pictures, Web-links, and even entire pages from the Internet.

The program is equipped with flexible sorting and filtering options, HTML-printing, automatic phone dialing, as well as integration with the installed e-mail client and Web browser. All traditional holidays are included in the program, plus you can add your own vital dates.

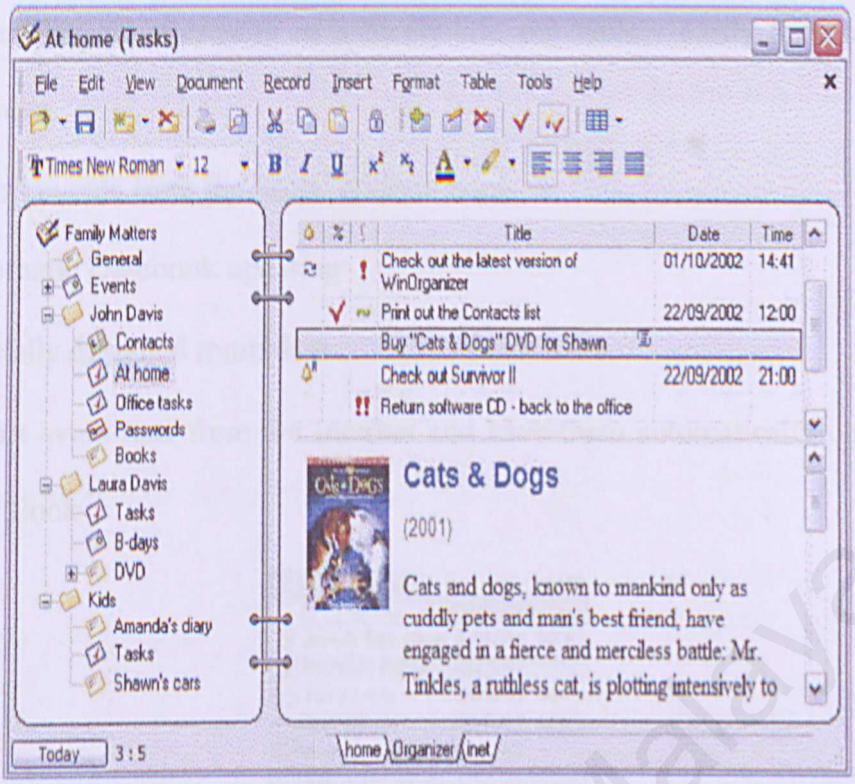


Figure 2.4: WinOrganizer

2.4.3 DateMate

DateMate is an organizer that arranges all the users' birthdays, anniversaries, business events, public holidays and any other events in one organized list. DateMate will automatically enter events in the users DateBook, notify them of up-coming events. Operating in Palm OS 3.5 or higher, this program is compatible to most of the Palm devices that are available in the market.

Features:

- With Zodiac view - view astrological information for each of the 12 zodiac signs.
- Handles any kind of event - from birthdays to community events to business events.

- All events are presented in a nicely-laid out table - easily browse the yearly calendar.
- Import events from the users' address book
- Automatic Datebook updating
- Specially designed reminders
- Import event lists from the Internet and have them automatically updated in the Date Book

Figure 2.5: DateMate

2.4.4 Comparison between existing software and Personal Organizer

The reviews among existing software solutions are as followed:

Existing Software Solutions	PalmForce Personal Organizer
MyWeek • MyWeek runs on Palm OS 3.0 or higher • Does not include notepad for user to jot down notes • Does not provide address book function for the users	• Personal Organizer runs under Palm OS 4.0 • Provides basic and sufficient functions that are needed by users for personal usage • Provides easy to use interface and reminder to alarm the users about certain events • Allow the users to jot down temporary notes • Allow users to set up alarm
WinOrganizer • WinOrganizer runs on desktop platform, that is the Windows OS • Includes a lot of personal management modules, allow users to manage their activities more effectively, but not suitable on Palm devices	
DateMate • DateMate runs on Palm OS 3.5 or higher • Does not provide notepad for users to jot down notes • Does not provide alarm clock functions • Good organize structure on events	

Table 2.4: Comparison between existing software and Personal Organizer

2.5 Chapter Summary

This chapter covers all the researches and studies that had been carried out to gain relevant knowledge on the project's criteria. Since I had no previous experience in developing a mobile application, I had search through the web to gather information in this area. These information had greatly aid me while I'm doing the latter part of this project.

The information that I had gathered include the programming languages that are available to develop a mobile application. The most popular are the C++ and Java. Therefore extensive studies had been carried out on this 2 criteria. Another important aspect is the type of the PDA that are available on the market today. This include the Palm device, Pocket PC and Psion. Their operating system, namely PalmOS, WindowCE and EPOC are being studied.

Another of research that had been carried out is the analysis on similar existing software. Programs that have the similar functions that would be the features in Personal Organizer are being reviewed and analyze.

After the literature review was carried out, a clearer picture and understanding of the project had been obtained. It is hoped that this review will help in the decision to choose the proper development tools and methods for the development phase which will starts in near future.

Chapter 3 Methodology

3.1 Introduction

The previous chapter have introduced the overview and information gathered in the literature search and review where as this chapter would further elaborate the justifications for the chosen project methodology, the reason why such methodology was chosen, and the fact-finding technique that is used to gather information throughout this project.

3.2 Software development life cycles

Software development life cycles are abstract model that define the activities, ordering of activities and information flow associated with the development of software. Life cycles exist for describing the entire development process as well as for particular focused activities such as system test or preparing to release software. These life cycle form the basis of project management. Without a life cycle to depict the activities that software development is supposed to undertake, a project management is not capable the software development has progressed.

3.3 Prototyping

Prototyping is a system development methodology that uses "trial and error" approach for discovering how a system should operate. The greatest strength of prototyping is that the process helps to develop a close working relationship between the system developer and the users. This relationship assists build trust and acceptance for the new system. A second strength of the prototyping is that it is arguably the best system development method for identifying how a system should operate when the system's specifications are hard to define. Also, the cycle of the prototyping process proceeds rapidly. Figure below shows how prototyping works.

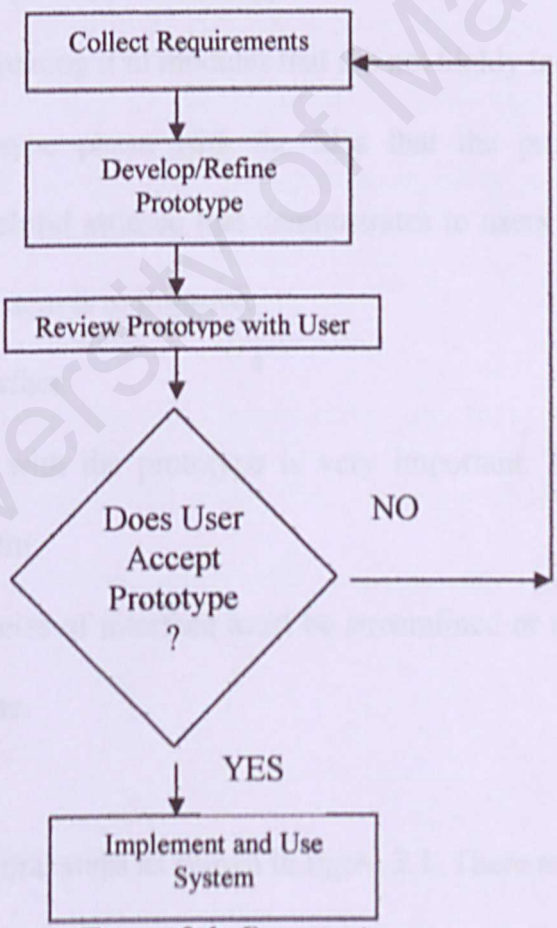


Figure 3.1: Prototyping

Once the decision to prototype has been made, there are 4 main guidelines that must be used when integrating prototyping into the requirements determination phase of the system. The guidelines are as follow.

- Work in a manageable modules
  - Speed is essential to the successful prototyping of an information systems
  - Putting together an operational prototype both rapidly and early in the system development allows the analyst to gain valuable insight into how the reminder of the project should go
- Modify the prototype in successful iterations
  - Construction of the prototype must support modifications. Making the prototype modifiable means creating it in modules that are not highly independent.
  - Entering the prototype phase with the idea that the prototype will require modification is a helpful attitude that demonstrates to users how necessary their feedback is if the system is to improve.
- Stressing the user interface
  - The user interface with the prototype is very important. For many users, the interface is the system.
  - Many of the intricacies of interface must be streamlined or ignored altogether in the prototyping phase.

Prototyping consists of several steps as shown in figure 3.1. There are

1. Identify Known Requirements
2. Develop Working Model

3. Use Prototype
4. Review Prototype
5. Repeated as needed

3.3.1 Identify Known Requirements

The determination of application requirements is as important under the prototype development method as when using the classical systems development life cycle method or structured analysis. Therefore, before a prototype is created, both user and system analyst work together to identify the known requirements that must be met. To do so, they determine the purpose that the system will serve and the scope of its capabilities. Throughout this and all the steps in the prototype development, many responsibilities are shared between user and analyst. In other words, the end-user is directly involved.

Activities in this phase consist of interviewing user management, summarizing the knowledge obtained, estimating the scope of the project, and documenting the result. User can have experience of organizing their daily life schedule from time to time thus providing a more systematic life style.

3.3.2 Develop Working Model

Prototyping uses an iterative development process. The process is started by developing a general plan so that individual knows what to expect from each other from

the development process. However, because of the nature of this development method, it is difficult to set a firm completion date. To begin the first iteration, user and analyst jointly identify the data that are needed in the system and specify the output the application must produce. This means describing the individual reports and documents the system should provide and the layout of each. The analyst also estimates the prototyping cost. The system analyst performs construction of the initial prototype. Speed in producing a running system is essential so that momentum on the project is not lost and so that user can quickly begin evaluating the application.

3.3.3 Use Prototype

It is the user's responsibilities to work with the prototype and evaluate its feature and operation. Experience with system in the actual application setting should provide the familiarity needed to determine what changes or enhancements are necessary or which in adequate or undesirable features to eliminate. [15]

During this stage, project must be created first and this project is original project and it is called prototype. By the way, this prototype is only use to modify.

3.3.4 Review Prototype

During the evaluation, there is a need to capture information on what users like and dislike, noticing why they react as they do. The information will influence the features the next version of the application should have.

It also provides insight into characteristic of the users and the business setting for the application. Changes for the prototype are planned before they are made.

3.3.5 Repeated As Needed

The process describe may be repeated several times to evolve the application. Four to six iterations are typical. This process ends when both users and analyst agree that the system has evolved to include the necessary features or when it is evident that there is no benefit to additional iteration.

During this stage, user can have more opportunity for changes in this system. If there are not any changes and additional features in this system, the process to develop the system is regarded as successful.

3.3.5 The Use of Prototype

Given below are the four ways to proceed after information gained from developing and using the prototype has been evaluated:

1. Abandon Application
2. Implement Prototype
3. Redevelop Application
4. Begin New Prototype

1. Abandon Application

Both prototype and application are discarded. Developing the prototype provided information from which to determine that the application or the intended approach is inappropriate to justify additional development.

2. Implement Prototype

The features and performance will meet user needs either permanently or for the foreseeable future. This strategy may be selected when the application environment is changing so fast that it is difficult to determine long-term or stable application requirements.

3. Redevelop Application

Development of the prototype provided sufficient information to determine the future necessary in the fully application. This information is used as the starting point for development of the application in a manner that makes the best possible use of resources.

4. Begin New Prototype

Information gained by development of the initial prototype suggests alternative strategies or circumstances. A different prototype is constructed to add to information about application requirements.

3.4 Reason why Prototyping model is chosen

1. Prototypes are inexpensive to build

The cost of building a prototype for PalmForce is relatively less than building a fully working application and this enable the project to be developed easily without much problem.

2. Fewer Errors

Prototype model can allow errors to be detected earlier. Requirements validation is performed since users are able to experiment with requirements and the system. The requirements validation process consists of 7 factors:

- correctness
- consistency
- completeness
- realistic
- needed
- verifiable
- traceable

3. More opportunity for changes

Some of the requirements proposed in this project earlier were change from time to time according the needs of each module and also because of the specific time frame given for it to be completed.

4. Unfamiliar with Palm programs' interface

Since Palm programs are quite new for me and I only have little knowledge and experience using it, then prototype is the best method to be used because it can help to reduce errors in the end of the product.

5. Designing a system for users' needs and expectations

Using prototype in development better fits with users' needs and expectations. A Palm Organizer mainly concentrates on the needs of the users especially those who want an application to help them to organize their daily schedule. Furthermore, being a user-

friendly application, this project has graphical user interface that is attractive, direct and easy to understand.

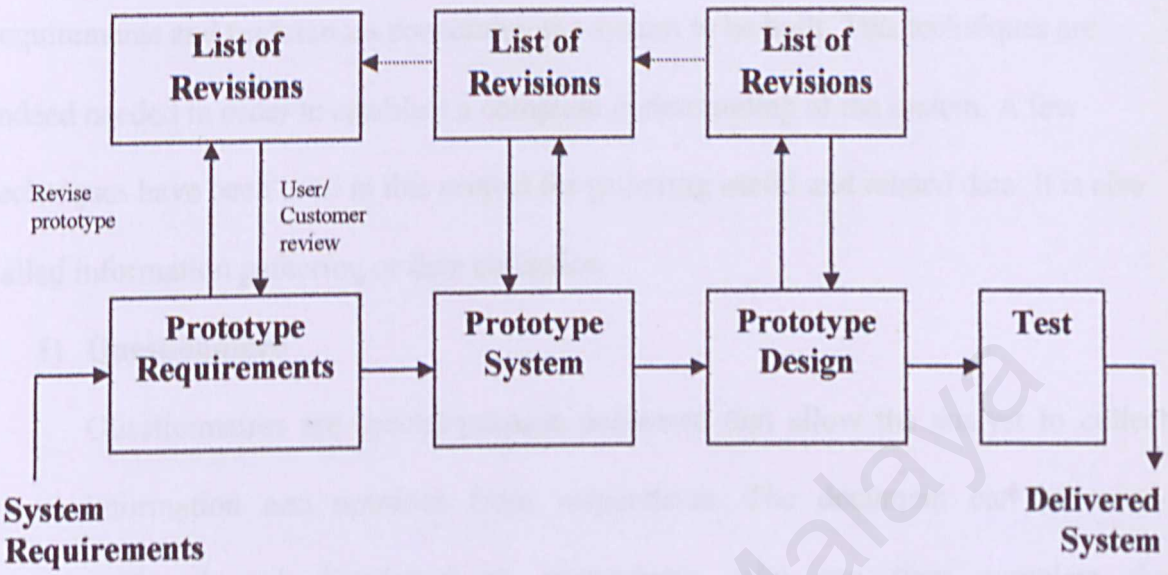


Figure 3.2: Prototyping model

3.5 Fact Finding Techniques

Fact-finding techniques refer to the methods of gathering information, requirements and preferences concerning the system to be built. This techniques are indeed needed in order to establish a complete understanding of the system. A few techniques have been used in this project for gathering useful and related data. It is also called information gathering or data collection.

1) **Questionnaire**

Questionnaires are special-purpose document that allow the analyst to collect information and opinions from respondents. The document can be mass-produced and distributed to respondents, who can then complete the questionnaire on their own time. Questionnaires allow the analyst to collect facts from a large number of people while maintaining uniform responses. When dealing with a large audience, no other fact-finding techniques can tabulate the same facts as efficiently.

2) **Internet Surfing**

Internet is used as the main resource for referring any ambiguities that arise during the entire development period. Through the Internet, some ideas from the similar system have been collected and the user interface design of similar system can provide the necessary guidelines. The Internet can also provide the technique on how to use the development tools in this project. All the knowledge about the hardware and software can be found in Internet.

3) **Interview**

Several informal interviews and discussions with the people who used some similar existing system have been carried out. These people include student, working people and etc. By interviewing these people, information on how people feel about the software and application for PDA on today's market can be acquired. Because that some of the existing system is used as a private and confidential system, thus informal interview will allow the developers to gather personal opinion of the users about such system.

4) **Documentation Study**

Documentation that is referred in this context is ranged from technical software papers to books on PDA programming. Certain journal and article about programming with PDA are gathered and studied to get a clear picture on how to properly develop a program for PDA users.

3.6 Chapter Summary

In order to develop a system that is really fulfilled the users needs, a proper development process model should be followed. The model should be chosen carefully, according to the needs of the development process and the unique features in the system.

This chapter reveals the methodology that had been chosen to develop the system. The model chosen is the prototype model which in short means developing the system by referring to the user feedback on the prototype. With this model, user will have more chances to view their opinion to the developers, thus allow the developers to change the system according to the user's needs. In addition, this model would be less cost expensive compared to other models. Errors can be detected in a much earlier stage during the development process so that it can be solve before it is too late.

In this chapter the information or fact finding techniques had been revealed as well. These techniques are used to gather valuable information about the system and ease the developers work on the requirements analysis. They also served as the knowledge portal where the developer find their solution through this technique when problems are met.

With the help of the prototyping model, it is hoped that the project will go on smoothly. The strong point of the prototyping model can help a lot to bring this project to success.

Chapter 4 System Analysis

4.1 Introduction

System Analysis refers to the investigation, analysis, design, implementation and evaluation of an information system, usually with the aim of computerizing some human activity. First, the problem to be solved must be accurately defined. The existing system is then investigated to understand how it works, using techniques such as flow charts and decision tables. Next, the results of the investigation are analyzed and used as the basis for the design of a new system, making optimum use to the available computer hardware, software, and staffing resources. If it is a new system to be developed, all the user requirements such as user interface, data entry screen, reports, security and etc as well as the possible constraints need to be identified until the complete understanding of the system is gained.

4.2 Questionnaire Summary

4.2.1 Questionnaire Design

Four important points to be considered in a questionnaire design are:

- The need to make things easy for the subject
- The need for unambiguous questions
- The need to gather precise data
- The need to support intended analysis

The combination of open-ended questions and closed question are chosen in the questionnaire design. Close question is chosen to limit the response options available to the respondent and eventually ease the analysis and interpretation of their responses without using a computerized content analysis program. Open-ended question is chosen to gather all possible responses to the questions from the responses. The design of the questionnaire is attached in *appendix*.

4.2.2 Questionnaire Result

Respondents

First of all, this survey which is carried out using questionnaires was conducted to know the user response and comments of Personal Organizer. The respondents were consists of local university students, college students, adults, profession, workers and many more. The questionnaire was distributed to the user of 100. This is to make sure that more accurate analysis can resulted from the questionnaires. Besides that, sufficient respondents are the main component in the survey

Result

The overall of survey was based on the questionnaires that collected from the user and data needed to be analysis to obtain the user requirements. Therefore, complete details of the questionnaires are important and some questionnaires have been discarded because of uncompleted form. Therefore, only 76 complete questionnaires gathered from the respondents where the remaining 24 questionnaires were discarded. The analysis result is shown as below:

	Existing System	Total	Percentage (%)
1.	Computer	38	50.0
2.	Special Book	8	10.5
3.	Personal Digital Assistant (PDA)	30	39.5

Table 4.1: Usage of Mobile device

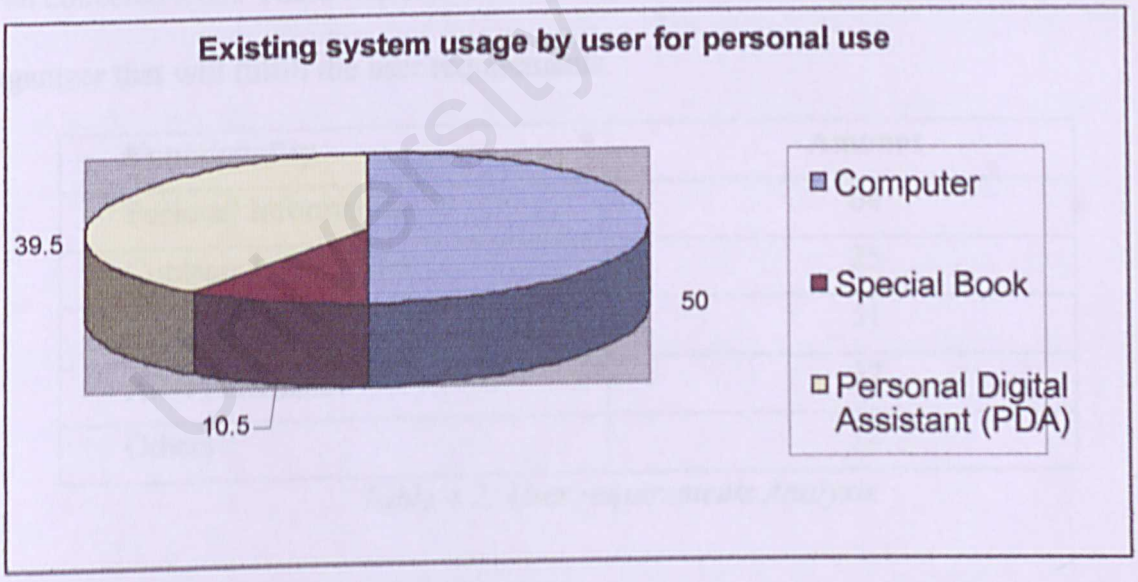


Figure 4.1: Existing mobile system usage

From the survey, it's clear that more than half of the total of user which is 50.0% uses the computer to manage their personal information. The user chose the computer to store personal information is that the computer storage is much greater than a handheld device like PDA. User would like to store more data into their own computer like personal computer and notebooks. The following method is to use PDA organizer in which 39.5% user would like to use a PDA to personal information storage. In the advancement of computer technology, the user felt that PDA would be a suitable device to store information. Since some of the user who inexperience in computer technology, only 10.5% of the users prefer to use special book.

User requirements for Personal Organizer

Through the questionnaires, user requirements for the Personal Organizer have been collected from. These requirements are important to build a robust Personal Organizer that will fulfill the user requirements.

Functionality	Amount
Personal Information	64
Contacts Information	25
To-Do Planner	51
Alert Functions	32
Others	12

Table 4.2: User requirements Analysis

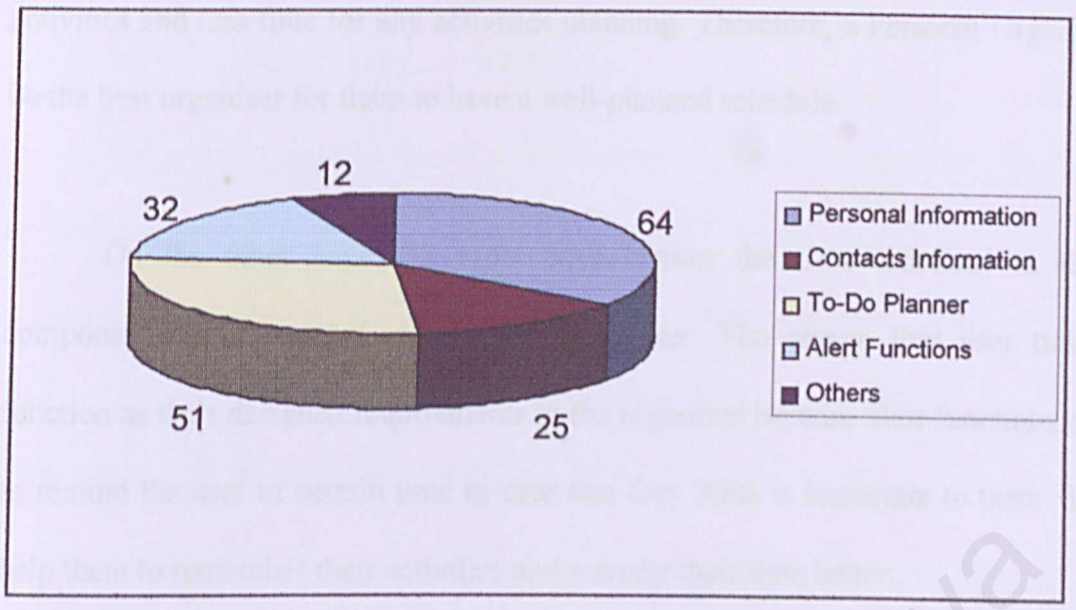


Figure 4.2: User requirements pi chart

Then, functionalities for the Personal Organizer that required by the user were collected from the respondents. This question was set to enable the user to choose the features or functionality that they would like to have in the Personal Organizer. Since this is a personal organizer, most of the user which is 64 out of 76 has commented that personal information must be included in the Personal Organizer. The main reason is they can store several of personal information such as weight, height and other information in their personal organizer. Data can be easily accessible and readable from anywhere and anytime.

The following highest total out of 76 users was To-Do planner where 51 users stated that they need to have some To-Do planner in their daily life. They need to have a planner to schedule for their daily activities. Nowadays, people are busy with daily

activities and less time for any activities planning. Therefore, a Personal Organizer will be the best organizer for them to have a well-planned schedule.

On the other hand, 32 users have chosen the alert functions to the third component that important in Personal Organizer. The reason that user ticked this function as their designed requirements in the organizer because alert functions can help to remind the user of certain time or date that they think is important to them. This can help them to remember their activities and manage their time better.

Finally, contact information is one of the functionality that required by the user which 25 user among 76 would like to store the contact information.

Conclusion

In conclusion, the questionnaires had successfully collected the user requirements for the Personal Organizer.

4.3 System Requirements Analysis

System Requirement Analysis is an activity that encompasses investigating a current system to fully understanding how it works, it present problem, existing constraints on the solution of the desired system [Barbara Robinson, Many Prior, 1995]. Soft system methodology has been adopted to identify problem situation, to analysis requirements and to design how system to solve problem.

Requirement analysis covers the area of functional and non-functional requirements of the Personal Organizer module. Anyway, this phase needs proposed users as well as the outline and the technology to be used in the system development are defined. It is indeed an important feature of the system or a description of something that makes the system capable to proceed while fulfilling the module's intention.

4.3.1 Functional Requirements

Functional requirements refer to the description of activities, functions and services or, features a system must provide. In these cases, the system should react to particular inputs and should behave in some particular situations. Sometimes, it is also stated that what system should not do. Furthermore, it is independent from the implementation of the solution.

Personal organizer is a program that allows the users to schedule their tasks and stores their contacts. It contains a reminder too that allow the users to set the PDA to display text or beep sounds to remind them of certain events. Below are the functional requirements for the personal organizer:

1) Setup Personal Information

The users can store their personal information with this function. The info includes name, date of birth, gender, address, phone number etc.

2) Edit Contacts

Users will be able to add and edit their contacts information with this function. The information that can be store includes the contact name, address and phone numbers.

3) View Contacts

Users can view the contacts they stored in their PDA in a complete list with this function. The information displayed is the contacts name and their phone numbers, in a list.

4) Edit To Do List

A To Do List is provided in this module to allow the users to write down the things that they want to do and remind themselves later with the reminder function of the system. Users can edit the list with the edit function.

5) View To Do List

The system will generate the To Do list in a proper format for the users to refer to. The list will be consisting of the things that need to be done, the date and the

time. Users can view the detail about the events in the list that they had selected too.

6) *Set Reminder*

User will be able to set the event in their PDA with reminder that would alert the user about certain events that need their attention.

7) *Set Alarm*

User will be able to set the alarm to the time they needed to alert themselves.

8) *View Time*

User will be provided a digital clock where the user will be shown the current time, date and day of the week on the screen.

4.3.2 Non-Functional Requirements

Non-functional requirements define the description of other features, characteristic, and system constraints that define a satisfactory system. It is a description of the constraints that limit the boundaries of the proposed solutions. The non-functional requirements for the system are:

- *Reliability*

  The system should be reliable in performing its daily functions and operations. It shall not produce any fatal or costly failures when it is being used. Besides, it is expected to perform its intended functions accurately and precisely.

- *Usability*

  The system should be easy to use and navigate. The complexity of the system shall be eliminated. The usability of the system will promise it will not limit and restrict the daily operations as well as the administration.

- *Security*

  The system ought to provide sufficient security to prevent leakage of confidential private information. When the users need to access such information, authentication of the identity of the users will be carried out.

- *Manageability*

  The system should be developed in the sense of easy manage. All the modules and functions integrated will be provided in simple mode. This will make the maintenance and enhancement simpler and less time consuming.

- *Flexibility and scalability*

  The system should be able to perform and executed in different platform. It will take advantage over the technology that is chosen to construct it and expand it to meet changing requirements.

- *Robustness*

  The system that is consisting of different modules will be testing to ensure each module achieves its expected outcome and functions. After the modules testing, the modules will be then integrated and tested. Any errant that is discovered during the system testing will be solved immediately. This will ensure the robustness of the system to its expected level.

4.4 Conclusion on Selected Tools and Technology

4.4.1 Developer platform – Windows XP

Windows XP is the latest release of Microsoft Windows operating system series. It includes the core modules that is being inherited through the Windows family, with enhancement in security and file management. It has great networking features, allowing user to set up a local network easily and transfer file between them. The remote control features is another strong point for Windows XP. The reasons why Windows XP is chosen as the development platform is that:

1) It provides compatibility to the development tools needed

   Windows XP allow developers to set up the development environment fast and convenient. The tools that are needed to build this project are all support by this version of Windows, so there is no worry on the compatibility.

2) Great support on networked development platform

   As mentioned earlier, Windows XP allow developers to set up their on local network fast and easy. They'll just need to install their network card on the workstation and plug them together. This allow file sharing can be done between the developers, which is an important features needed in this project.

3) Good security features

   Allow developers to protect their file from unknown users.

4.4.2 Programming Language – Java

Java 2 Mobile Edition(J2ME), as stated in chapter 2, is a Java platform release by Sun Microsystem to meet the demand of mobile programming. It includes classes and packages for programmers to use when developing programs for mobile devices. There are several reasons why it is chosen as the main technology used in this project:

1) It is free

   Java provides the J2ME as a free source to download from its web site. Developer can go to the company's web site to download the package and the relevant information about it.

2) Can be developed and run on multi platform

   One of the special features of Java technology is that it can be run on different platform without modification on the program, as long as the platform is installed with the required virtual machine. J2ME is no exception. Developer can choose to develop on a Linux based platform or a Windows based platform, and the application that is built can be run in either one operating system too.

3) It is supported by most of the world's leading manufacturers for mobile devices(PDA)

   The advancement of Java technology has been noticed by a lot of mobile device manufacturer in recent years. Today a lot of mobile device, including PDA, support Java programs and applications. These manufacturers include Palm, Sony, Nokia, etc.

4) Java programs and application is getting more popular

The popularity of Java programs and application is getting more popular among PDA users, especially among the youth. Java built programs come in small sizes and run effectively, suitable to the environment of a mobile device.

5) Less development time

Java classes' structure allows developers to reuse them once they had been instantiated. This can help developers to save time by using these existing classes.

4.4.3 Development Tools – Borland JBuilder 9.0

Borland's JBuilder 9.0 is an Integrated Development Environment (IDE) specially tailored for Java developers. . It comprises almost everything that is needed by the developers, including the compiler and the debugging mode. The user interface is fairly easy to use, and it fully support mobile edition. The Java Wireless Toolkits 1.0.4 is included in the package, therefore it ease the developer job to find another suite for the wireless toolkit. The main reason why it is chosen as the IDE for this project are:

1) It includes all the necessary components needed to develop this project

There are several extra components that are needed to develop this module properly, and with JBuilder 9.0, everything is in a bundle. Components such as the virtual machine, editor, debugger and plug-in for the Palm emulator are included in the IDE package. Thus it is very suitable to be used in this project.

2) It has full support from Borland

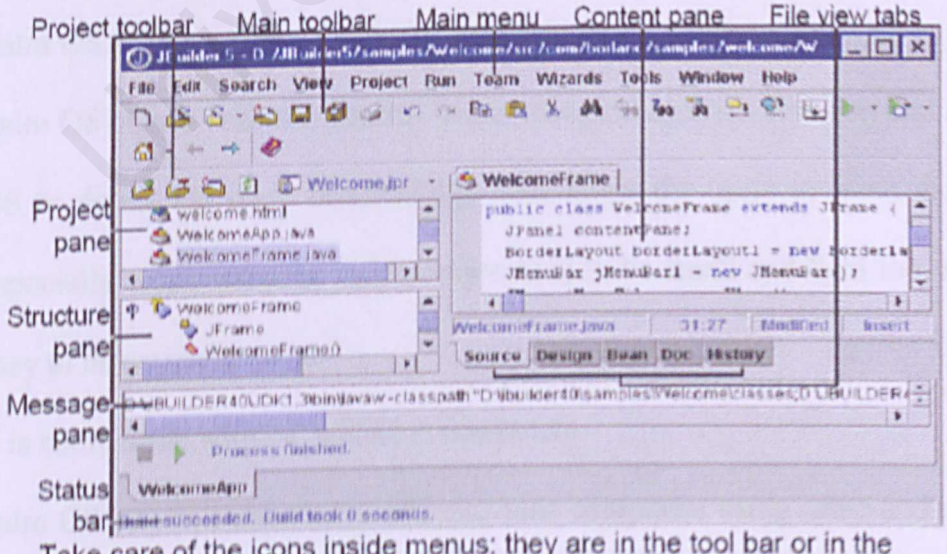
Being developed by Borland, it receives full support from the company. Updates and FAQ can be found on the company's web site and can be downloaded easily.

3) It has easy to use interface and features to help developers

Borland's JBuilder comes with a well designed interface that allows developers to browse through their files easily. Every command and function is well place for easy reference. The editor can displace the possible function and parameters within a class, similar to the Virtual Studio release by Microsoft. JBuilder 9.0 can automatically generate useful codes for the developers as well when they want to start a new project, thus saving work and time for the developers.

For personal usage, Borland's JBuilder can be downloaded for free from the following web site: http://www.borland.com/jbuilder/

Below is demonstration of the IDE's interface:



Take care of the icons inside menus: they are in the tool bar or in the message pane… Hot keys are given too.

Figure 4.3: JBuilder user interface

4.4.4 User Platform – PalmOS

Palm OS, as stated earlier in chapter 2, is a powerful operating system developed by Palm Inc that comes with all the basic functions that are needed to perform on a PDA. There are several reasons why Palm OS is being chosen as the target platform, listed below are some of them.

1) It is the most famous PDA operating system

   Palm OS is one of the most famous PDA operating system in the world today. Although it had been challenged by various type of operating system, Pocket PC for example, but it has improved itself and still leading the operating system sales.

2) Lots of models that meet different users demand

   Palm OS PDA comes in various model, some with extensive multimedia function, some are well designed for business people. Thus it can attract users in various field.

3) Palm OS PDAs are small, light and easy to use

   Palm OS comes with the Grafitti handwriting recognition function that allow the OS to detect the users handwriting. This allow the users to input data easily, especially while walking. Palm device tend to be small and light too, making it easy to be carried around.

4) It is compatible with PC and Mac computers

   Palm OS can be connected to PC and Mac computers using cable and a conduit. Data can be exchanged between the desktop and the PDA without much trouble.

4.5 Chapter Summary

In this chapter the summary of the information gathered is being displayed. Functional and non-functional requirements are being listed down in order to give a clear picture about what the system should do. In the functional requirements list, the requirements are divided into the general requirements and specific requirements. For the non-functional requirements, there are a few important aspect being discussed, including the security, usability, robustness, manageability, reliability, flexibility and scalability. Then the software and hardware requirements for both the user and the developers are being discussed.

After working on this chapter, a clear set of functions and requirements are being defined. This can aid us to work on our next chapter, system design.

Chapter 5 System Design

5.1 Introduction

System design is the specification or construction of a technical, computer-based solution for the business requirements that being identified and established during the stage of system analysis. In other words, the design phase addresses *how* technology will be used in the system . It focuses on the technical or implementation concerns of the system. Therefore, design phase is concerned with technology-based views of the system's data, processes, and interfaces.

System design is important in serving the purposes as stated below:

- To transform requirement into a working system.
- To determine a set of components and intercomponent interfaces that satisfies a specified set of requirements.
- To change the abstract logical model to the concrete physical implementation.

5.2 Structure Chart of Personal Organizer

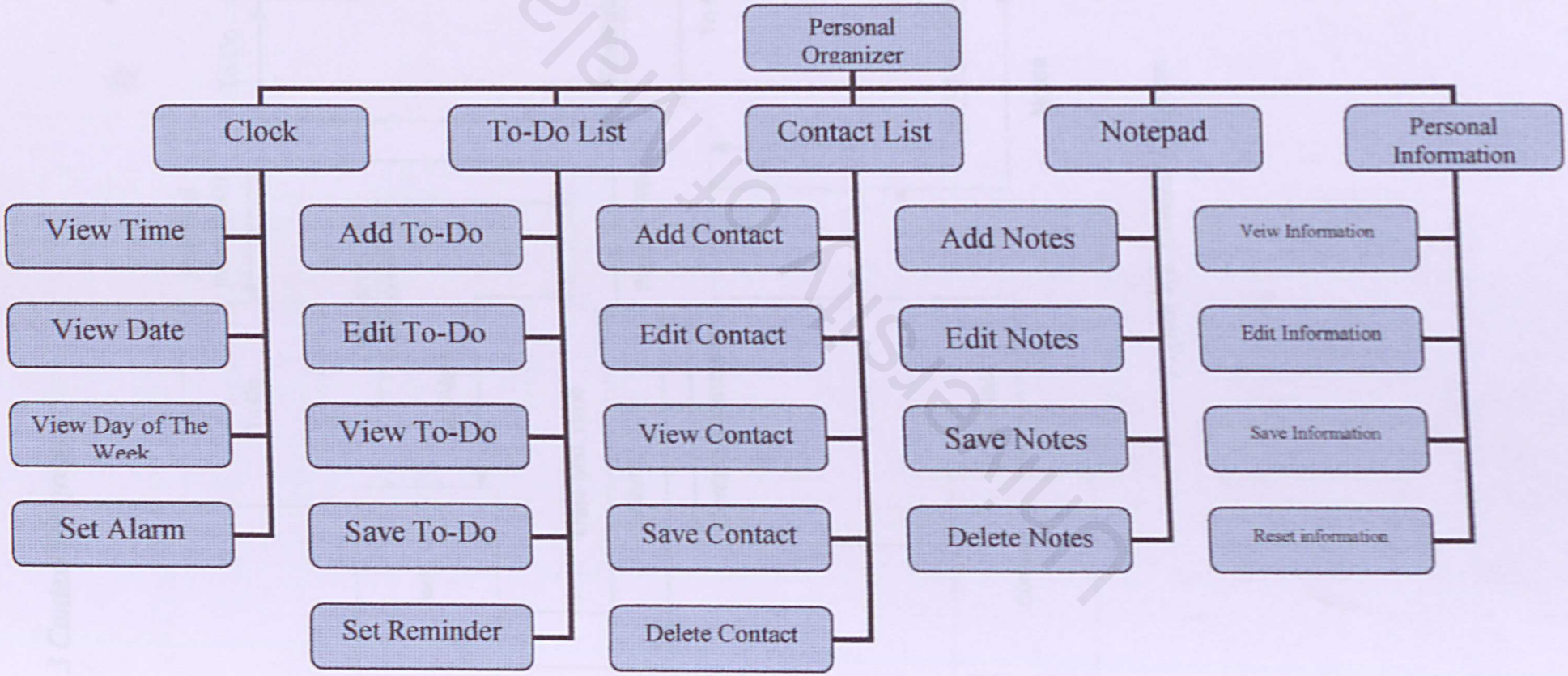


Figure 5.1: Structure Chart for Personal Organizer

5.3 Context Diagram

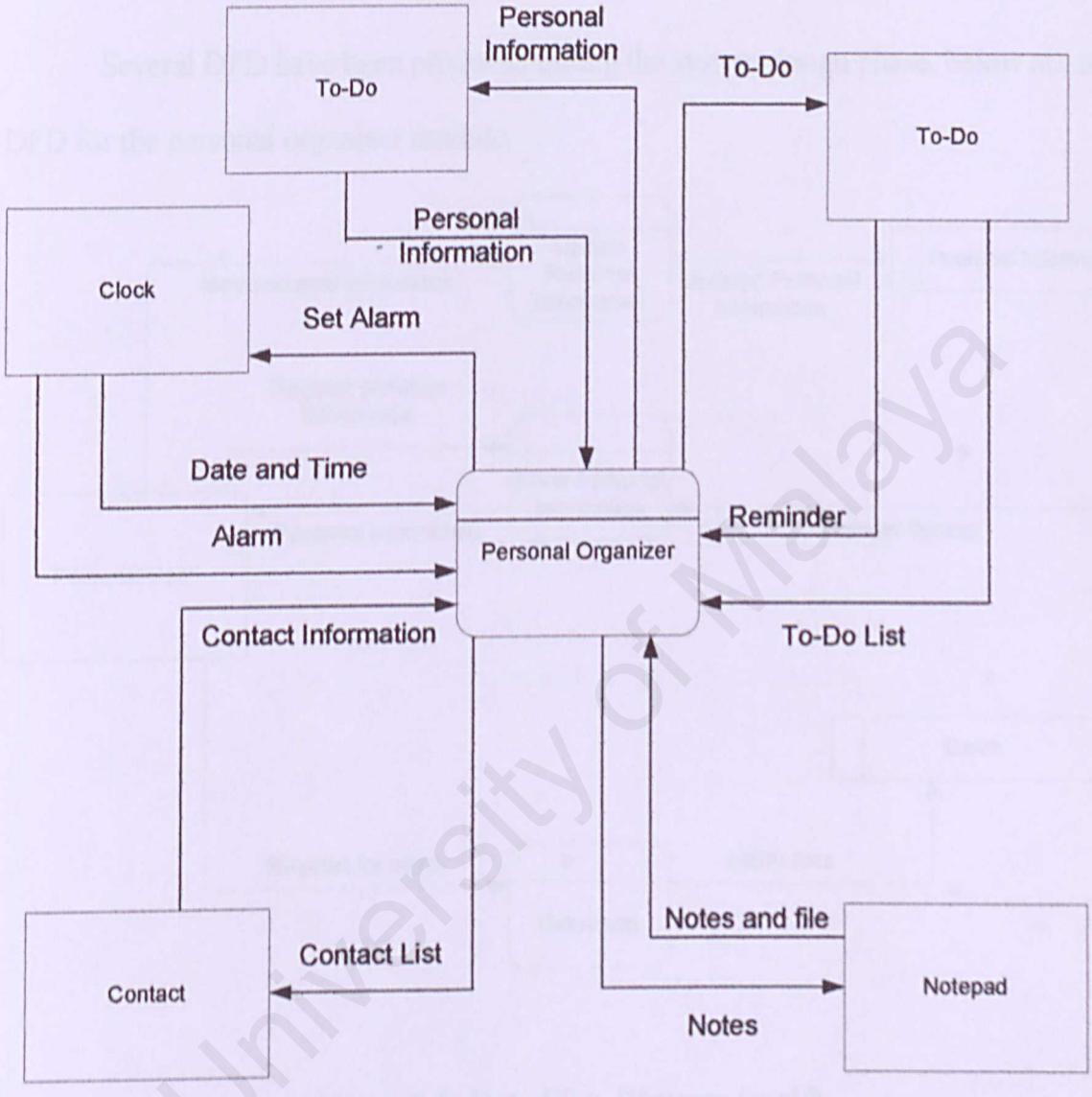


Figure 5.2: Context Diagram

5.4 Data Flow Diagram

5.4.1 Data Flow Diagram level 0

Several DFD have been produced during the system design phase, below are the DFD for the personal organizer module.

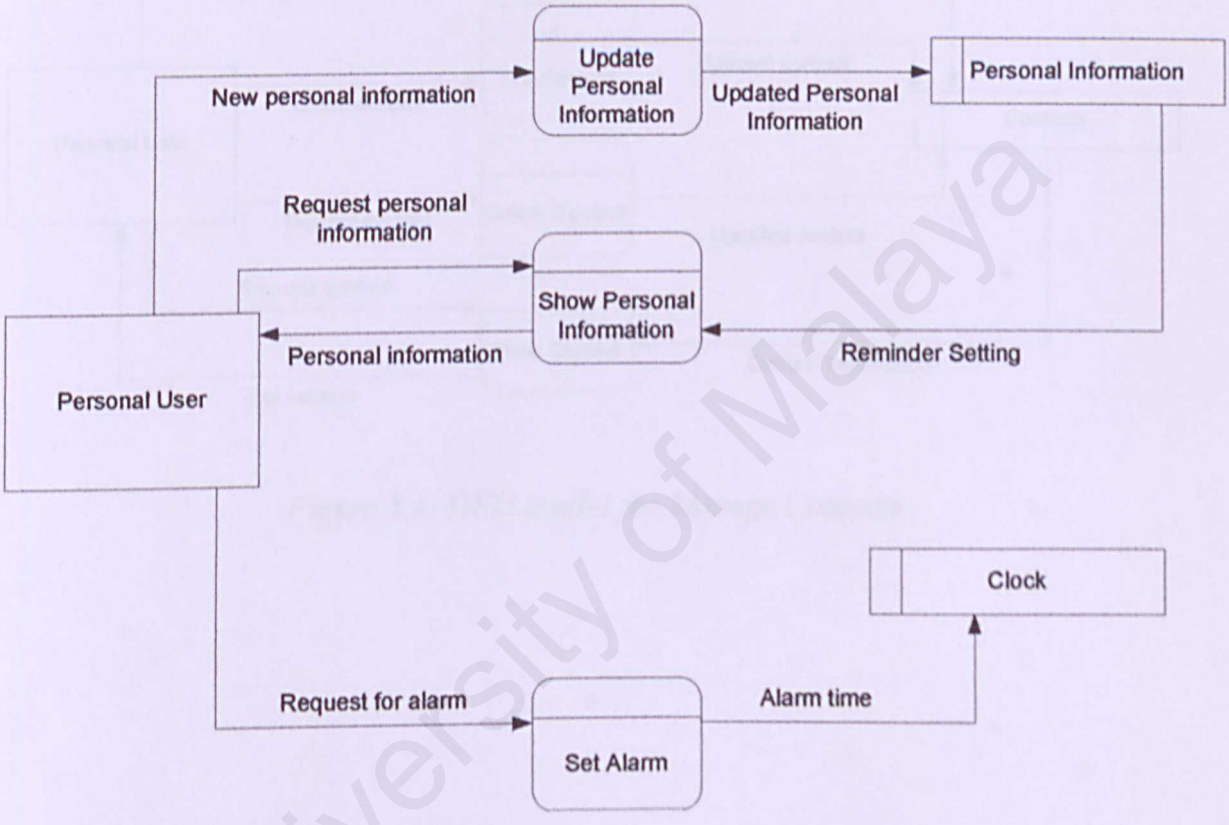


Figure 5.3: Data Flow Diagram level 0

5.4.2 Data Flow Diagram level-1 for Manage Contacts

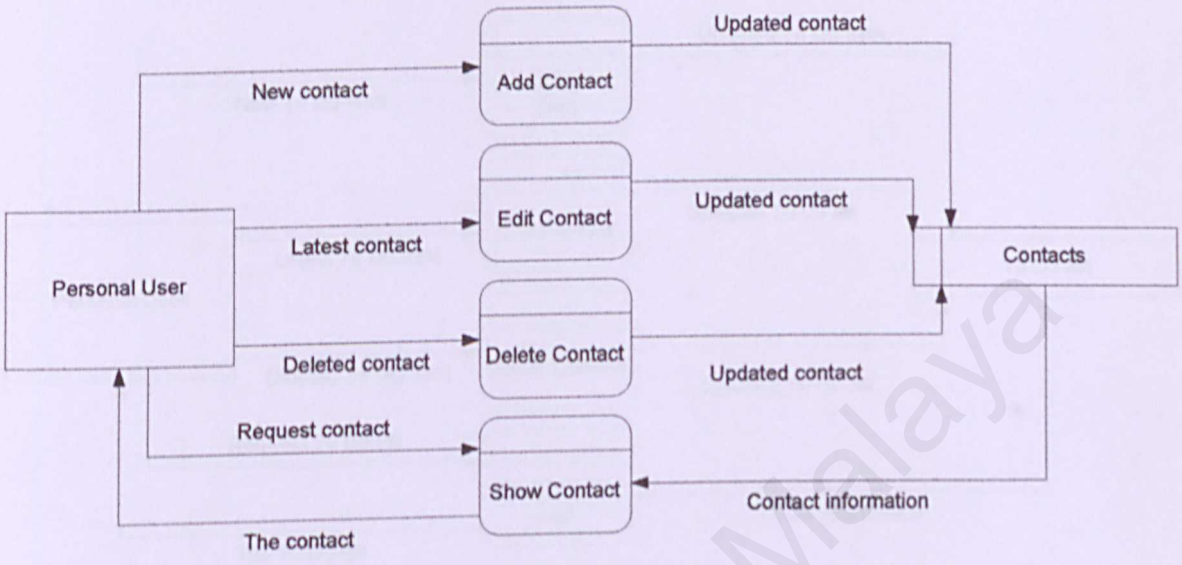


Figure 5.4: DFD level-1 for Manage Contacts

5.4.3 Data Flow Diagram level-1 for Manage To Do List

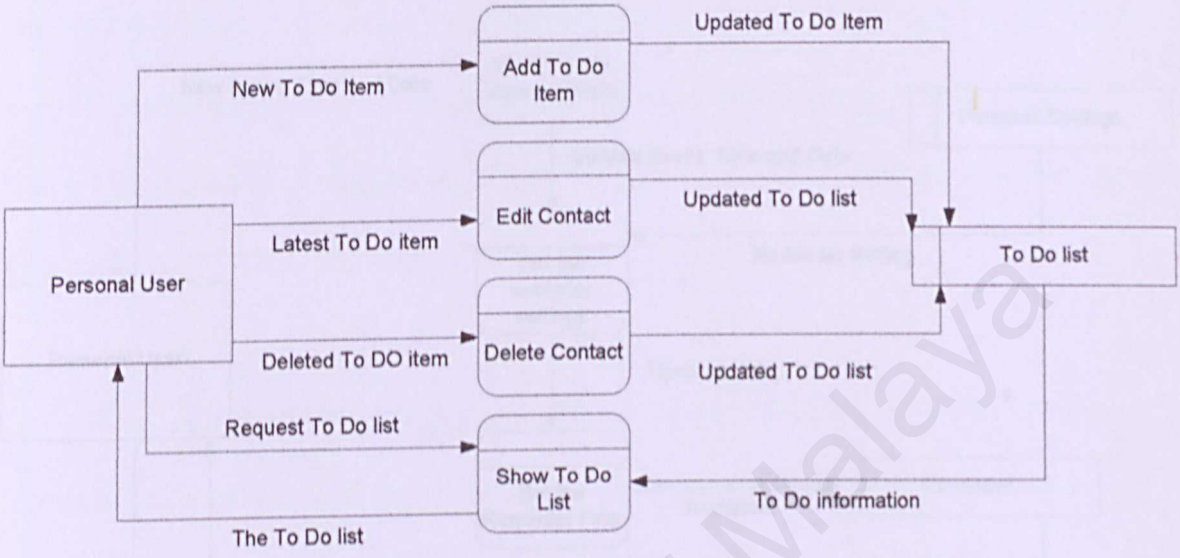


Figure 5.5: DFD level-1 for Manage To-Do List

5.4.4 Data Flow Diagram level-1 for Control Reminder

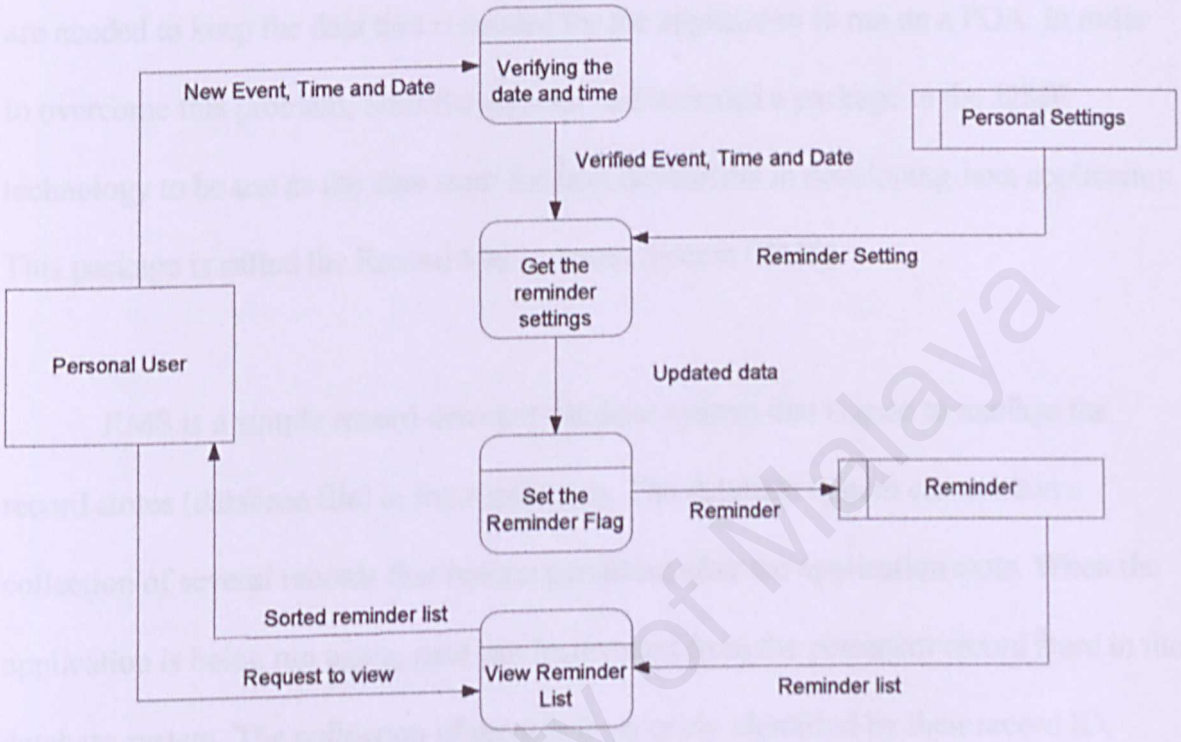


Figure 5.6: DFD level-1 for Control Reminder

5.5 Database

Application development on a mobile device faces a lot of constraint. One of them is the storage problem. Memory on a PDA is very limited, thus special 'databases' are needed to keep the data that is needed for the application to run on a PDA. In order to overcome this problem, SunMicrosystem had included a package in the J2ME technology to be use as the data store for Java developers in developing Java application. This package is called the Record Management System (RMS).

RMS is a simple record-oriented database system that is used to manage the record stores (database file) in the application. The database system can contain a collection of several records that remain persistent after the application exits. When the application is being run again, data can be revoked from the persistent record store in the database system. The collection of records is uniquely identified by their record ID, which is an integer value. The data for Personal Organizer is store in several record stores. Below is the basic design for them.

1) **Reminder**

   This record store is being used to keep the reminder information that Personal Organizer receive from the user. The data kept in here are the reminder status ,the event that needs to be reminded, the event's time and the date.

Data field	Data type	Description
Status	Boolean	Set to true if the reminder is active, otherwise false
Event	Varchar	The location of the event in the record store
Date	Date	The date of the event
Time	Time	The time of the event

Table 5.1: Data in the Reminder record store

2) Notes

This record store is used to keep the notes made by the user. These short notes then can be viewed by the user later. The data stored here include the notes written and the file name,

Data field	Data type	Description
Notes	String	The notes
File Name	String	The file name

Table 5.2: Data in the Notes record store

3) Personal Information

This record store contains the personal information of the user. All the details about the user are being kept here. The data field kept here includes the user name, date of birth, address, home phone number, mobile phone number ,password, height, weight, and account balance,

Data field	Data type	Description
Name	String	The user name
Date of birth	Date	User date of birth
Address	String	The user address

HomePhoneNumber	Long	The user home phone number
MobilePhoneNumber	Long	The user mobile phone number
Password	Varchar	The password for login
Height	Integer	The user height
Weight	Integer	The user weight
AccountBalance	Long	The user account balance

Table 5.3: Data in the PersonalInfo record store

4) Contacts

This record store is used to store the information about the user's contacts. User can input the information from the screen and recall it later. The data stored include the contacts name, contacts address, contacts phone number and contacts date of birth,

Data field	Data type	Description
Name	String	The contact name
Address	String	The contact address
Phone1	Long	The contact phone number1
Phone2	Long	The contact phone number2
Date of Birth	Date	The contact birthday

Table 5.4: Data in the Contact record store

5) Alarm

This record save the time that user set the alarm. The only field in this record is the time that the user set.

Data field	Data type	Description
Alarm	String	The alarm timer

Table 5.5: Data in the Alarm record store

6) To-Do

This record store keeps the To-Do-s that the user keeps in the PDA. The data kept include the To-Do event, the date and the reminder flag,

Data field	Data type	Description
Event	String	The To-Do event
Date	Date	The date for the to-do list
Remind	Boolean	Set to true if the reminder is set for this activity

Table 5.6: Data in the ToDo record store

5.6 User Interface Design

User interface design is concerned with the dialogue between a user and the computer. It is concerned with everything from starting the system or logging into the system to the eventually presentation of desired outputs and inputs. Therefore, user interface design is aimed to help users and businesses get the information they need in and out of the system by addressing the following objective:

- ***Effectiveness*** as achieved through the design of interfaces that allow users to access the system in a way that is congruent with their individual needs.
- ***Efficiency*** as demonstrated through interfaces that both increase the speed of data entry and reduce errors.
- ***User consideration*** as demonstrated in the design of suitable interfaces and by providing appropriate feedback to users from the system.

- ***Productivity*** as measured by ergonomically sound principles of design for user interfaces and workspaces.

Since the methodology that is being used to develop this system is the prototype model, a user-interface prototype has been created using Visual Basic to allow the user to participate in the process to determine the interface design. The following are some of the screen shot from the prototype for the Personal Organizer module in the project.

5.6.1 Screen Shot from the prototype for Personal Organizer

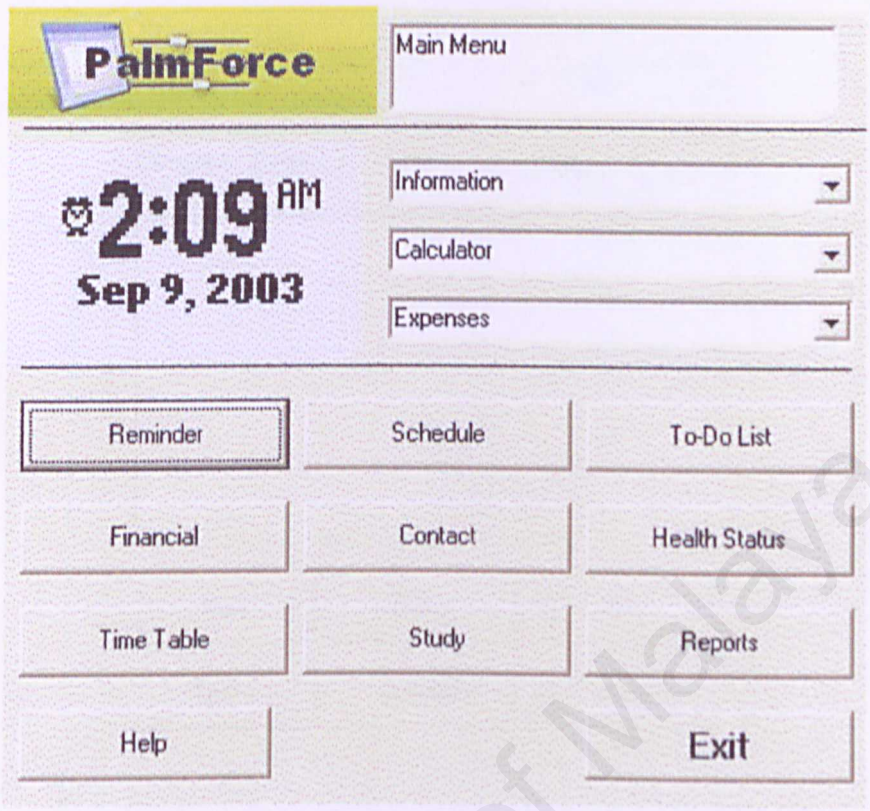


Figure 5.7: Main screen of prototype

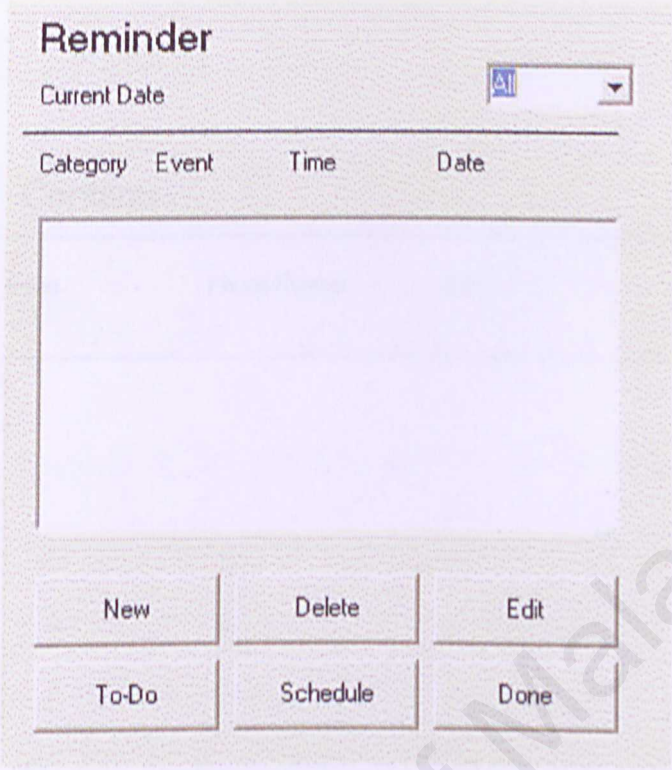


Figure 5.8: Reminder of Personal Organizer prototype

Figure 5.9: To-Do list of Personal Organizer prototype

Figure 5.10: Contact of Personal Organizer prototype

5.7 Chapter Summary

System design is defined as those tasks that focus on the specification of a detailed computer-based solution, on the technical or implementation. There are three kinds of design showed in this chapter, which are the design diagram, the database and the user interface design.

The design diagrams in this chapter consist of the structural chart, the context diagram and the data flow diagram. These diagrams partition the system into subsystem and functions. They show the overall of the system to be developed. Data flow diagram provides a clear outline for the developer to understand how data would be transfer in their modules. Database design is the process of translating logical data models into physical database schemes. Since this application is going to be use on a PDA, thus its database needs special design as well. The record store design allows the developers to store data persistently on the device while saving space. The record store that is going to be used in this system is discussed here too. User interface design is concerned with the dialogue between a user and the computer. Too complicated user interface will prevent users from using the system. Some of the user interface from the prototype is being displayed here.

Finally, it is hoped that the system design defined in this chapter will provide a full development guide in the development phase later.

Chapter 6 System Implementation

6.1 Introduction

The requirement analysis, methodology and system design phases do not have a clear boundary in a software project. Each phase tends to overlap one another. System implementation is a process that converts the system requirements and design into program codes.

6.2 System Development

System development consists the used of methodology chosen, forms coding, development tools, database connection and configuration on server. The details are illustrated as below:

6.3 Methodology

This project will be developed using prototyping model. The development phase that would be carried out in this project are requirements analysis, prototype system, prototype design, and testing. Each of the prototyping phase will be revised until satisfaction. After that evaluation is done on the final product. Documentation is being carried out throughout the project as well.

6.4 Development Environment

6.4.1 Hardware Environment

As stated earlier in chapter 4, there are a certain hardware specification that need to be met in order to develop this module. Below are some a list of the hardware specification that is being used during the development process:

Computer Processor Unit	Processor not less than 1GHz
Hard Disk Space	10.0GB
Random Access Memory	256MB RAM
Additional devices	Printer, USB port, Serial port, CD-RW
Operating System	Microsoft Windows XP Professional

Table 6.1: Hardware development environment

Note that there are changes in the list compared to the one in chapter 4. This is due to the increase demand on the processing power on the workstation as the project being run. When the emulator is being run on, the workstation would need extra processing power and memory to run it smoothly. Therefore the demand is increased. Another additional device is the CD-RW. During the development process there is a demand to copy files from a workstation to another, and since the volume of the floppy drive and USB hand drive is limited, this a cd-rw is needed to record these files.

6.4.2 Integrated Development Environment (IDE)

The chosen IDE that would be used to develop this system is Borland's JBuilder 9.0. It comes together with the Java Wireless Toolkit 1.0.4, which is the main Java package that would be used to develop application for mobile devices. JBuilder 9.0 is

chosen as it makes the integration and coding progress simpler and easier. As other usual IDE, it provides the function to:

- Language aware editing
- Project definition facilities
- Integrated compilation
- Integrated stepwise execution (debugger)

JBuilder 9.0 can integrate with different types of emulator fairly easy as well. With simple wizard and form, the emulator can be set up and test the application once it is build. This feature allows the developer to test the application, the prototype especially, in a convenient way. The debugger is very helpful as well. It would automatically detect some of the error during code writing stage, so that the developer would know the error before compile it.

6.4.3 Emulator

An emulator is a program that mimics a device operating environment, so that when a user need to do something with this, he/she would just need to install the suitable emulator on a PC and do the work. To test the codes for this program easily, an emulator is needed to allow the codes to be executed on the workstation. Since this module is develop for Palm OS, thus an emulator for the Palm devices is being set up in the workstation as well. This emulator can be set up in JBuilder. Together with the emulator is a set of ROM that is required in order to run the emulator.

6.4.4 Graphic Editor

Graphic editor such as Adobe Photoshop and the ACDSee is being used in the development process, since there are icons involve in the module. Images used in a application for the Palm device must be in PNG format. Therefore a graphic editor is needed to change the format of the image into PNG format so that it can be used in the module.

6.4.5 Text Editor

Text editor is needed as documentation is done throughout the development process. Microsoft's Word 2003 is being used for documentation in this project. Other text editor such as the Notepad in the Window XP and Adobe Acrobat reader is being used as well to view some of the Java reference.

6.5 System Coding

Coding is an iterative process whereby it is done until the programmer obtains the desired results. There are two types of coding approach; one is top-down and the other one is bottom-up. The bottom-up coding is based on coding some complete lower level modules and leaving the high level modules merely as skeletons that are used to call the lower modules, whereas the top-down approach is the reverse.

For this system, coding is done with the bottom-up approach. The advantages of this approach are: testing can be carried out on some of the functions as soon as it is completed, and critical functions can be coded first to test their efficiency.

6.6 Coding

6.6.1 Database Coding

The database used in J2ME is the rms package that comes together in the java wireless toolkit package. It is a database style that uses stack technique to store data in the device. In order to save space and avoid redundant, my group member and I had written a class that utilize the rms function and provide a few other functions that would allow us to control the database in a simpler way. The basic functions include the open function, where a database is being prepared to be accessed. The code for this function is as follow:

```
try {
    this.rs = RecordStore.openRecordStore(recordName, true);
} catch (Exception e) {}
```

The try and catch segment is required to catch the exception being thrown out by the function. Another function is the add data function. This is a function where we had modified to suit the need of our application. Below is the code for this function:

```
public void addData(String data) {
    byte[] nData;
    nData = data.getBytes();
    try {
      this.rs.setRecord(this.currentFld, nData, 0, nData.length);
    } catch (Exception e) {}
  }
```

The parameter *data* is the input from the application that need to be store in the database. This function would prepare the data in a stream of bytes form, and use the *setRecord* function to store the data in the database.

Interface coding

The interface components provided by the Java package is limited. Therefore in order to display our data in a systematic output, a class called myTable is created to display the data in a table format. This class would allow the developer to display at the maximum 3 column of data. Color would be added to each row. Below is a few lines of codes used to draw up the table:

```
void paint(Graphics g){
g.setColor(150, 0, 0);
g.fillRect(0, 0, 160, 15);
}
```

The first code is used to set the color of the graphic object to the desired color. And the second code is used to draw a rectangle with the color. Another important graphics object that is being used in interface coding is the image object. Most of the icon used in the module is being displayed by using the image object. Below is a sample line of code using this object:

```
img = Image.createImage("/Icon/clock.PNG");
```

This code would create an Image object *img* from the file clock.PNG. After this, the image can be display in the interface by using the following code:

```
g.drawImage(img, 2, 25, 0);
```

This function would draw the image stated earlier form the position (2,25). The width and height of the image would be determine by the size of the image itself.

6.7 Chapter Summary

When a developer come into the implementation phase, the developer have to decide how to write the code and develop a functional system that fulfills the design requirements. For this project, the situation is the same. The coding approach chosen for develop this module is the bottom-up approach. The development methodology is revised to ensure that the implementation process is done properly.

Before starting to do the coding, the development environment must be set up properly first. Therefore in this chapter the hardware and software requirements that is needed to develop the module is being list out to use as a reference. The hardware include the workstation specification and the peripheral device needed. The software required include the correct operating system, the IDE, the emulator and the assisting tools such as the graphic editor and the text editor.

After the development environment is set up, then the coding process starts. The core function for this module is being reviewed throughout the coding progress so that the relevant and critical function can be done properly.

Chapter 7 System Testing

7.1 Introduction

Testing is an integral component of the software process and an activity that must be carry out throughout the life cycle. During the requirement phase, the requirements must be checked; design phase requires careful checking as every other phase. During coding, each part of the application must be tested and the product as a whole needs testing at the integration phase. In other words, it is not sufficient to test end product merely at the end of the software development life cycle.

Testing is not the first place faultfinding occurs; but testing is focused on finding faults, and there are many ways to make the testing efforts more efficient and effective.

7.2 Testing Techniques

There are two type of testing technique applied in the testing stage of the system: white box testing and black box testing.

7.2.1 White Box Testing

White box testing is a testing case design method that uses the control structure of the procedural design to derive test cases. Using white box testing, test cases with the following characteristics can be derived:

- Guarantee that all independent paths within a module have been exercised at least once.

- Exercise all logical decision on their true or false sides
- Exercise on al loops at their boundaries and within their operational bounds
- Exercise on internal data structures to ensure their validity

7.2.2 Black Box Testing

Black box testing focuses on the functionality requirements of the system. It enables the developer to derive sets of inputs conditions that will fully exercise all functional requirements from an application. Black box testing was not used as an alternative to white box testing technique rather this technique is used as a complementary approach that is likely to uncover a different class of errors. Black box testing attempts to find errors in the following categories.

- Incorrect or missing functions
- Interface errors
- Errors in data structures or external data access
- Performance error
- Initialization and termination errors

7.3 Testing Strategies

There are three types of testing, namely, unit testing, module testing and integration testing. After a program is completely coded, it will be tested under unit testing. Module testing will start when all the programs under a particular module have been completely coded and tested under unit testing. The integration testing is to recover errors associated with interfacing when integrating all the modules.

The objective of testing is to find error and fault. Fault identification is the process of determining what fault or faults caused the failure, and fault correction or removal is the process of making changes to the system so that the faults are removed. There are several types of faults where it can be categorized as algorithmic faults, syntax faults and documentation faults. Algorithmic fault occurs when a program algorithm or logic does not produce the proper output for a given input because something is wrong with the processing steps. Syntax fault can be checked while parsing for algorithmic faults. This will ensure that the construct of programming language is used properly. Documentation fault occurs if the documentation does not match what the application does, and such faults can lead to other faults later because of the wrong implementation.

7.3.1 Unit Testing

Units testing focus on verification effort on the smallest component of the system design. Each component is treated as a standalone entity and tested individually to ensure that they operate correctly. The unit test is usually white-box oriented and the step can be conducted in parallel for multiple components.

Testing can start by examining the program code by reading through it, trying to spot algorithm, data and syntax faults. Test also can be performed by comparing the code with the predefine specifications and design to ensure that all relevant cases have been considered. Finally, test cases are developed ensure that the input is properly converted to the desired output.

Following steps are used in carry out the unit testing for the Personal Organizer:

- The code of the program is examined by reading through it to spot for possible algorithm, data and syntax faults.
- Control objects are tested to ensure its functionality.
- Different data types are used to test the error handling function.
- Test cases are developed to ensure that the input is properly converted into the desired output.

Examining Source Code

In this stage, the codes of the program are read to identify faults. After that, a code walk-through is carried out. In a walk-through, the code and the accompanying documentation are presented to the review team. Then, the team will comment on their correctness. For this project, the review team members consist of my course mates. Walk-through is conducted in an informal manner. This method is useful to identify faults that have been left out by the programmer.

Control Object Testing

Command buttons are clicked to test their functionality and text boxes are tested with different data types and also null value to make sure invalid data will not cause any fault.

Different Data Type Testing

Different data types like numbers, characters or date is used to test certain function because some control objects will only accept certain data type, invalid data type can be traced by the system without causing any error.

Unit Test Case

Below is a list of test cases used in Unit Testing

	Test Procedure	Expected Output	Test Result Analyzing
1	Attempt to fill up character in the IC field for Personal Information	Module reject the input	The module detected the fault input. Receive only numeric input
2	Set the date in the form and refresh the form	Module should be able to display the correct date	The correct date is able to be displayed
3	Change the content of a note and click to save it	Module should detect that it is an existing notes and save the content to the correct file	Content of the desired file updated. Module able to detect the modified file
4	Select a file and open the file	Module search out the file and open the note	The selected file is open. Module able to detect the selected file
5	Set the alarm to a certain time and save	Module should be able to 'remember' the time that had been set	The form display the correct time. Module able to remember the correct time
6	Enter long data in the event field for To-Do list	The table should be able to put dot at the behind of the shorten sentence.	Table display the event with dots at the behind of the sentence. Able to shorten long statement

Table 7.1: Unit test case

7.3.2 Module Testing

Module testing is to test the form of the system. All the programs under a sub-module are grouped into one form and all the related forms are grouped into a module. This testing will make sure all the forms are link to the right location.

Module Test Case

	Test Procedure	**Expected Output**	**Test Result Analyzing**
1	Enter and save data on the contact list form. View the result in the table	Module should be able to collect the data and display it on the table	The correct data is being display on the table. Module able to get the correct data from the database
2	Click the back or main or exit button on each form	The desired form should be loaded to the screen	Module able to load the desired screen. All the forms are linked together correctly
3	Click the save and open button in the notes form	Module should be able to display the correct form	Each of the form is loaded correctly

Table 7.2: Module test case

7.3.3 Integration Testing

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with the interfacing. The objective is to take unit tested components and build a program structure that has been dictated by design. This testing will ensure that the section in this project, which are call module, to be arranged correctly.

The approach used in this phase is an incremental strategy, the bottom-up integration and regression testing. The incremental integration is the antithesis of the high bang approach. This web application is constructed and tested in small increments where errors are easier to isolate and correct. All the interfaces are tested completely and a systematic test approach is applied.

For this project, a bottom-up approach has been used. Bottom-up integration testing begins construction and testing with modules at the lowest levels of the system and then moving upward to the modules at the higher levels of the system. Regression testing is the re-execution of some subset of tests that already been conducted to ensure that changes have not propagated unintended side effects. It is the activity that helps to ensure that changes (due to testing or for other reason) do not introduce unintended behavior or additional errors.

At the culmination of integration testing, this project is completely assembled as a package, interfaces and linking errors have been uncovered and corrected. A final series of software tests that is the validation testing has been carried out during this phase.

Software validation is achieved through a series of black box tests that demonstrate conformity with requirements. For this project, a test plan has outlines the classes of tests to be conducted, and also a test procedure is used to defines specific test cases that will be used to in an attempt to uncover errors in conformity with requirements. Both the plan and procedure are designed to ensure that:

- All functional requirements are satisfied

- All behavior characteristic are achieved
- All performance requirements are attained
- Documentation is correct
- Other requirements are met (e.g. error recovery, maintainability, compatibility)

Integration Test Case

	Test Procedure	Expected Output	Test Result Analyzing
1	Set the alarm and wait until the time comes	Module should be able to display the alert box on time	Alert box was displayed on time. Module able to retrieve the correct time and check the time with the current time
2	Set the reminder in the To-Do list and wait until the time comes	Module should be able to display the event on time	Alert box pop up on the selected time. The event string is able to be displayed in the alert box.
3	Click "New" button on the Notes and write notes Save the file and open it again	A new page should be provided to type message. The file can be save with correct file name	The correct content is being displayed. Module able to provide a new page for notes writing and save it in database
4	Click on the row for Contact table	The correct information of the selected record should be displayed	Successfully display the correct content in the edit page
5	Click on the row for the To-Do list	The correct information of the selected record should be displayed	Successfully display the correct content in the edit page

Table 7.3: Integration test case

7.3.4 System Testing

Final testing procedure done is system testing. However, testing the system at whole is very different from previous unit testing and integration testing. The objective

of unit testing and integration testing is to ensure that the code has implemented the design properly. In other words, the code is written to do what the design specifications intended. In system testing, a very different objective is to be achieved, that is to ensure that the system does what the users want it to do.

There are several testing scenario exists. Yet, this system was tested with Performance Testing and Function Testing.

Performance Testing

Performance Testing addresses the non-functional requirements of the application. Once the functions are convinced work as specified, the performance test compares the integrated components with the non-functional system requirements. The types of performance tests carried out for this application are:

i. **Compatibility Tests**

This test was performed to find out that the interface functions perform according to the requirements. Results clarify that the accuracy of data retrieval was high in this system.

ii. **Security Tests**

This test ensures that the application fulfils the security requirements. Only the valid users are granted the access to the secure zone.

iii. **Volume Tests**

The fields and records are checked to see if they can accommodate all expected data.

Function Testing

Function testing is based on the system functional requirements. In other words, a function test is used to check that whether the integrated system performs its functions as specified in the requirements. Each module involved is tested individually to determine whether the system performs as required.

7.4 User Testing

Although all testing discussed earlier have been performed, another important testing will be user testing. End user will be the user that will be using this module. So, user testing is important to get a view of this module and any comment from them is useful to enhance this module based on their requirements.

The users that helped to test my module are our housemates and coursemates. Another friend of mine, Mr. Hoong Chee Hoe, who's currently studying in USM, had helped to test my module as well. Below are some of the comments that are given by the user:

- The module is easy to understand
- Necessary and useful functions are included
- The module is slow when running on the real device
- The user interface is user-friendly
- Several Bugs found
- The module is convenient to use

7.5 Chapter Summary

This chapter starts with introducing testing to the reader. Testing is done after coding and implementation process. The purpose of testing the module is to find out faults that are undetected during coding implementation process. Besides that, testing helps to verify functionality of each of the module according to the system design. Different types of testing were carried to test the system so that it is reliable up to certain standards.

The testing process that had been carried out on this module is being discussed in this chapter as well. The testing includes the unit testing, module testing, integrating testing and system testing. The method used is described with relevant test cases.

Chapter 8 System Evaluation

8.1 Introduction

When the system is fully tested for errors, the next process would be the system evaluation phase. The evaluation process will be able to identify the system strength, system limitations, and future enhancements. The problem encounter and the technique that is being used to solve it is evaluated as well. There are many evaluation techniques that can be used to evaluate the final system. The following sections would discuss about the evaluation results of this system.

8.2 Problems Encounter and Solutions

From the initial development phase until the end of development phase, many problems were faced. Among the encountered problems, some can be solved by using certain solution while some remain unsolved or could not be solved due to a few factors such as hardware limitation, lack of resources. The following are some of the problems that arise during the development process.

i. **Lack of the knowledge in mobile application development**

Since I have no previous experience in developing a mobile application, I had no knowledge about how a program is written for mobile devices such as a PDA. Therefore in order to solve this problem, a lot of researches and readings had been done to gain information on mobile programming. Internet resources are the most helpful in the information gathering process. Some of the specific question

is being post on the forum to gain opinion from other mobile application developers as well. Relevant information about J2ME programming is gathered as well, since it is the main programming language used in this project.

ii. **Difficulty in finding a suitable emulator**

The emulator for Palm devices are quite difficult to find. My group member and I had gone through several web sites but still failed to find it. After some additional research and using the advance search of the search engines, we finally found one suitable copy of emulator in the Palm Developers web site.

iii. **Lack of knowledge in choosing IDE for mobile application development**

There are various type of IDE available in the market nowadays for mobile application development. But to choose a right type of IDE was a difficult task for me as I lack of the knowledge about the IDE available. Each of the IDE is specially design for certain programming language usage. For Java development, the available IDE are such as Sun Microsystems SunOne studio, Borland JBuilder and IBM WebSphere. To solve this question, relevant information about each of the IDE is being gathered and discussion is held between my group member and me. After discussion we had agreed to use the Borland's JBuilder as out development IDE.

iv. **Difficulties in coding with J2ME**

Due to that J2ME is a different platform that is used for mobile programming, I had problems when trying to code the functions for my module. Although the syntax is the same as the Java I had learnt before, the functions and API for each of the statements are different. In order to solve this problem, the API documentation is being downloaded for reference. Several article and tutorial

related to J2ME had been read to gain the necessary knowledge in mobile programming using Java. Discussion with my group members about the coding problems helps to solve my problem as well.

v. **Coordination among the development group members**

Since this module is going to be integrated with other module to form a complete system, it is very important for us to communicate and discuss the standard and methods used in the modules. To save space and practice function reuse in our system, some of the classes are being shared among us to use. Therefore we need to update ourselves of the changes that are being made to these classes. To allow better communication we had linked our workstation together and create a folder where our source codes would be stored. Then we can update our codes through this folder. Discussion is another method used to exchange opinion between us.

8.3 Systems Strength

The strength of this module is listed as follow:

i. **Provide convenient and useful organizing tools**

This module would provide useful tool for its user to manage their daily activities and store their personal information.

ii. **Easy to use interface**

The user interface designed for this module is quite simple and straight forward. This allows the user to learn this module quickly and master its usage in a short time. The user interface also allows the user to use it easily and without hassle. Usage of color in the interface makes the screen more attractive as well.

iii. **Provide interacting tools for the user**

This module would allow the user to set alarm and reminder to remind themselves of certain event. This allow interactivity between the user and the module. This service would features in other modules in the system as well.

iv. **Implement error handling**

During the coding of this module event handling and error handling codes are written as well. For Java, the most obvious error handling event would be the try and catch function that is required when coding some part of the module, especially when dealing with the database. This feature would allow the error to be detected easily.

v. **Convenient information management tool**

The information management method used in this module is systematic and clear on sight. The display of the data stored is performed by using tables and forms. These allow the user to read their data in a glance and reduce the need to click on the device.

8.4 System Constraints

Due to time and knowledge limitation, there are a few system constraints showed at below. This include

i. **Slow response time**

There are a few functions in the module that would be executed once the system is on. These includes the alert and reminder functions. Furthermore Some of the function such as the contact list and to-do list requires database loading. All these

functions would slow down the process of the module. Therefore the response time might be slow for certain Palm device.

ii. **Lack of synchronization with PC**

When the database or files in the module increases, the users can upload the database to their PC, but the synchronization of the data is hard to be done since there are no specific conduit for the module.

iii. **Confusion of time zone**

Certain device of Palm had the ability of changing the time zone of the system time. This module does include a class for handling time objects, but changing of time zone might result in wrong time and date from the module.

iv. **Depending on the device system and power**

Like any other application, this module is highly dependant of its operating system and device power supply for execution. If there is a shortage of power supply, then this application cannot be executed well.

v. **Alert and reminder function cannot be executed without the start of the module**

This means that in order to allow the module to check the reminder and alarm set, the users must execute the module manually when they switch on their device. This is due to the fact that Palm OS does not allow develop to access its API call function. Therefore the module cannot be auto-executed in the operating system.

8.5 Future Enhancement

System development is a dynamic process and changes must be expected. Due to the limited resources that I have, especially time, this has caused me have missed or overlooked certain aspect of the system. However, after the development system has been completed, I have identified certain important aspects that I can add on or modify for future enhancement.

Below are some of the additional features that can be implementing if time is given. The future enhancements features showed as below:

i. **Adding a draw pad**

During the development of the module I had tried to develop a draw pad where user can draw string or pictures on it. But it will took a long time in order to make the tool functional. Therefore it was depleted half way. Given time I believe that this tool can be completed.

ii. **Fix the time bug**

The time function for the J2ME is quite confusing when implemented in time zone enabled device, as mentioned earlier. Perhaps this problems can be solve with more efforts in research and coding in the future.

iii. **Enhance the user interface**

The user interface of the module can be enhanced with more detailed design and usage of color. Skins of the clock can even be provided, if the design is done properly. Therefore it is an aspect that can be improve in the latter version.

8.6 Knowledge and Experience Gained

The experience and knowledge gained while conducting this project are undoubtedly valuable. The knowledge gained will sure prove to be useful in the future. Below are some of the knowledge and experience I gained from this project:

i. **Enhance my Java programming skills**

I had learnt more about Java programming when I work on the coding of this module. It allows me to experiment the programming concepts I had learned in the faculty and finds out my mistake.

ii. **Knowledge about mobile programming**

This is no doubt the biggest harvest that I gained from this project. Before the project I know nothing about mobile computing. After working on this project, I managed to learn a bit of here and there in the mobile programming world, especially the J2ME platform

iii. **Improve Project and Time Management**

By working with my group members to build up the system, I had learned how to communicate in groups for developing a system effectively. We had learned how to manage a project so that it could be completed within the deadline as well.

8.7 Chapter Summary

The Personal Organizer Module for PalmForce application has been completed successfully, with some strengths as well as limitations as mentioned. The module has achieved and fulfilled the objectives and requirements, as stated in the proposal and system analysis.

However there are still spaces for improvement in the Personal Organizer module. The enhancement of the module can be made in the future to meet the changing needs of the users.

Finally this project has achieved its objective in giving the undergraduates an opportunity to undergo different challenges in different phases of the system development, that include research, system planning, system analysis, system design, system implementation and system testing.

Reference

[1] http://www.palmone.com/us/products/compare/

[2] http://www.ericlindsay.com/

[3] Comparison - White Paper - PalmOS vs. PocketPC by Eric Dampierre

[4] J2ME - www.developer.com/java/j2me/article.php/1475521

[5] Configuration, Profile, Optional Package – http://java.sun.com/products/cldc/

[6] J2ME table – http://java.sun.com/j2me/docs/j2me-ds.pdf

[7] J2ME - www.developer.com/java/j2me/article.php/1475521

[8] J2ME - www.developer.com/java/j2me/article.php/1475521

[9] J2ME - www.developer.com/java/j2me/article.php/1475521

[10] KVM - J2ME Building Blocks for Mobile Devices—May 19, 2000 – chapter 4 Introduction to the KVM

[11] Benefit of CLDC - J2ME Building Blocks for Mobile Devices—May 19, 2000 – chapter 1 executive summary

[12] Java vs c++ - Java vs. C++ on Palm OS By Theodore Beisler

[13] Conduit – http://palm.objectwareinc.com/

[14] Gary B.Shelly, Thomas J.Cashman, Judy Adamske, Joseph J.Adamski (1991) *System analysis and Design*, Boyd & Frase.

[15] James A.Senn (1989). *Analysis and Design of Information Systems*.2nd Edition, Mcgraw.Hill Publishing Company.

Appendix

Questionnaires

USER FEEDBACK QUESTIONNAIRE

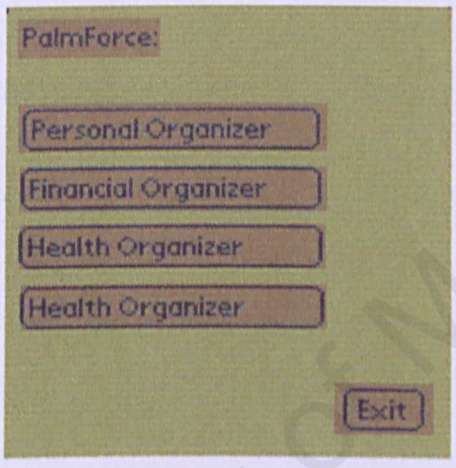
PALMFORCE APPLICATION

Name: _____

Date: _____

QUESTIONNAIRES FOR PROTOTYPING MODEL

The picture below is the main menu for the initial prototype for PalmForce system where it contains of four organizer such as personal organizer, financial organizer, health organizer and personal organizer. Each of the buttons will link to the each organizer system relatively. There is an "Exit" button for the user to exit from the PalmForce system.



1. By clicking on the button of each organizer you should be able to get their description. Do you think that this is a good way of obtaining the information

   YES/NO

   Comments:

2. Do you think the pictures on the following buttons explain their function?

   YES/NO

3. Do you think that the main menu for the PalmForce needs to have more buttons besides the exit button? You can choose more than one answer:

☐ "Help" Button

☐ "Setting" Button

☐ "Prescriptions" Button

☐ "Next" Button

4. Do you think that we need to include "Back" button for every organizer or after the user enter into each organizer?

☐ Very Agree

☐ Agree

☐ Disagree

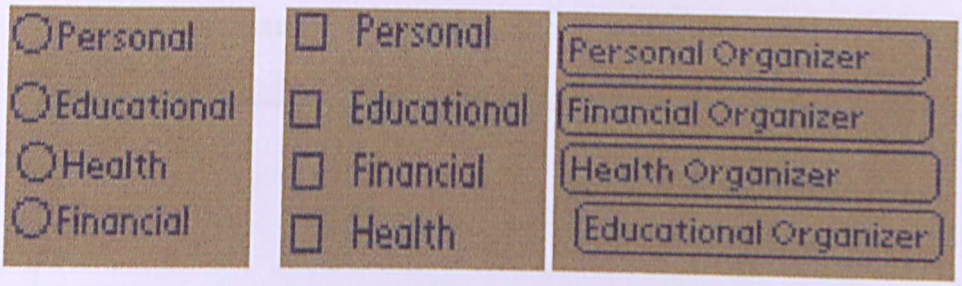
☐ Very Disagree

5. Referring to the prototype above, do you think that the text description for each button is suitable? For example, do you prefer to show "Health" instead of showing "Health Organizer"? Please choose one of the relevant box:

☐ Financial, Health, Educational, Personal

☐ Financial Organizer, Health Organizer, Educational Organizer, Personal Organizer

Comments:



A. Radio Box B. Check Box C. Button

6. According the picture above, which king user interface do you prefer, A, B or C. Please choose one of the options.

☐ A

☐ B

☐ C

7. Do you wish to have the above system which it is consists of several kind of organizer in the same system? Please give your comments.

Comments:

8. Do you think that information was presented in a well-organised way?

YES/NO

Comments

9. If NO can you please provide some feedback of how do you think it can be improved?

10. What did you think of the overall user interface? Please select one of the following:

- ☐ Bad
- ☐ Average
- ☐ Good
- ☐ Excellent

11. What could be improved?

Thank you for answering my questionnaire

USER FEEDBACK QUESTIONNAIRE

PERSONAL ORGANIZER

NAME : _____

DATE : _____

☐ Please tick in the relevant box

___ Fill in with the appropriate answer

1. How do you manage your daily important personal data or information?

☐ Personal Computer

☐ Special Book

☐ PDA

☐ Others, please specify: _____

2. Which is the fastest way for you to retrieve information if you own the following devices?

☐ Computer

☐ Special Book

☐ PDA

3. Do you feel that Personal Organizer will help you in organizing your daily activities? Please tick one of the boxes below.

☐ Very agree

☐ Agree

☐ Disagree

☐ Very disagree

4. What functionality would you expect from a Personal Organizer system? You can choose more than one answer.

☐ Personal Information

☐ Contacts Information

☐ Schedule Planner

☐ To-Do List

☐ Others, please specify: _____

5. Do you think that Personal Organizer will help you in managing your daily information?

☐ YES

☐ NO

6. What do you see as the strengths of using Personal Health Organizer to improve your health performance?

☐ YES

☐ NO

IF YES, PLEASE SPECIFY:

_____

7. What do you see as the **weaknesses** of using Personal Health Organizer?

☐ YES

☐ NO

IF YES, PLEASE SPECIFY:

_____

8. Please state briefly how you feel about using of this Personal Health Organizer.

COMMENTS

9. Any other comments?

COMMENTS

10. Would you like to use Personal Organizer if you have opportunity to have one?

☐ YES

☐ NO

Thank you for answering my questionnaire

PalmForce
Personal Organizer
User Manual

Table of Content

List of Figures

1.0 Introduction

Thank you for choosing PalmForce as your Palm's organizing tool. PalmForce includes four major organizing criteria which we believe would suit your needs well. This organizing tool includes a personal organizer, a financial planner, a healthcare manager and an educational manager.

This user manual would provide guides on how to install PalmForce on a Palm device and how to use the Personal Organizer included in PalmForce.

1.1 Overview of Personal Organizer

Personal Organizer includes tool for you to organize your daily activities with the easiest way. It consists of 5 major functions, which are:

1) Clock
2) To-Do list
3) Contact list
4) Notepad
5) Personal Information

Each of functions will be discussed in the following sections.

2.0 Installation of PalmForce

Before installing PalmForce on the device, users must install the Java Virtual Machine on the device first. Below is the installation guide for installing the JVM and PalmForce on a Palm device.

2.1 Installing Java Virtual Machine

Before installing any file, the Palm Conduits must be is installed first on the desktop. Palm conduit, which is called HotSync, comes together with the device.

The Java Virtual Machine used in Palm device is called Java ™ HQ. This file can be is named "MIDP.prc". Find the file and double click it. A screen as below should pop up.

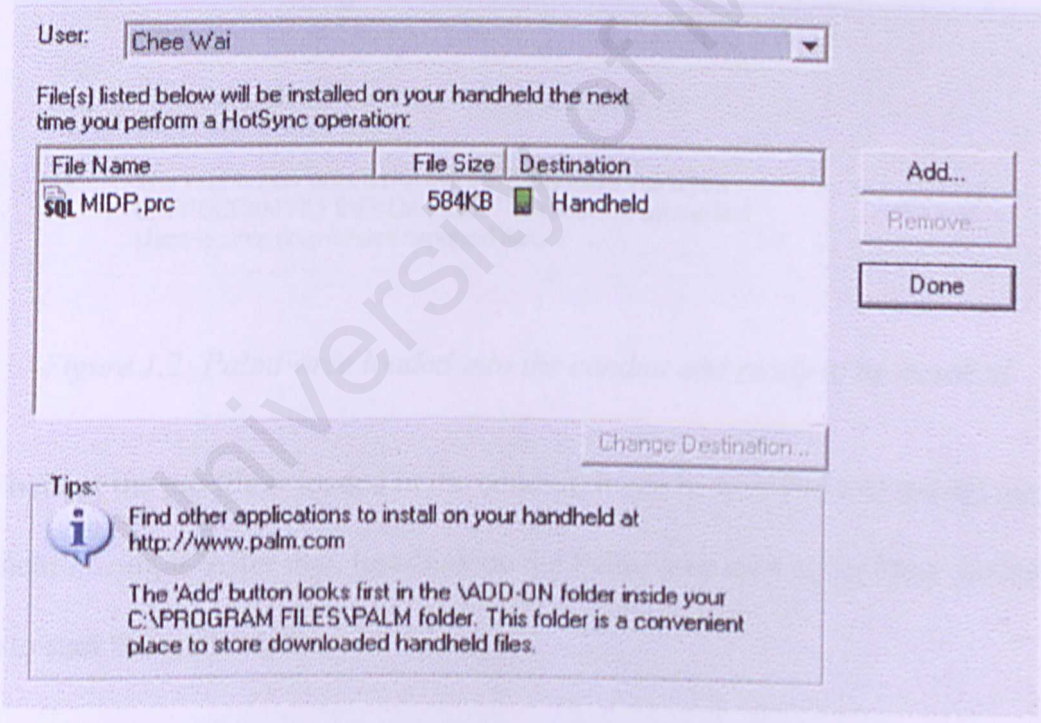


Figure 1.1: Loading JVM into the conduit

When the screen pop up with the MIDP.prc in it, the file should be ready to be loaded into the device. The PalmForce application file then can be installed.

2.2 Installing PalmForce

Continue from the previous installation, the PalmForce application can be loaded into the conduit by double click on the PalmForce file, which is "palmforce.prc". The conduit screen should show like this.

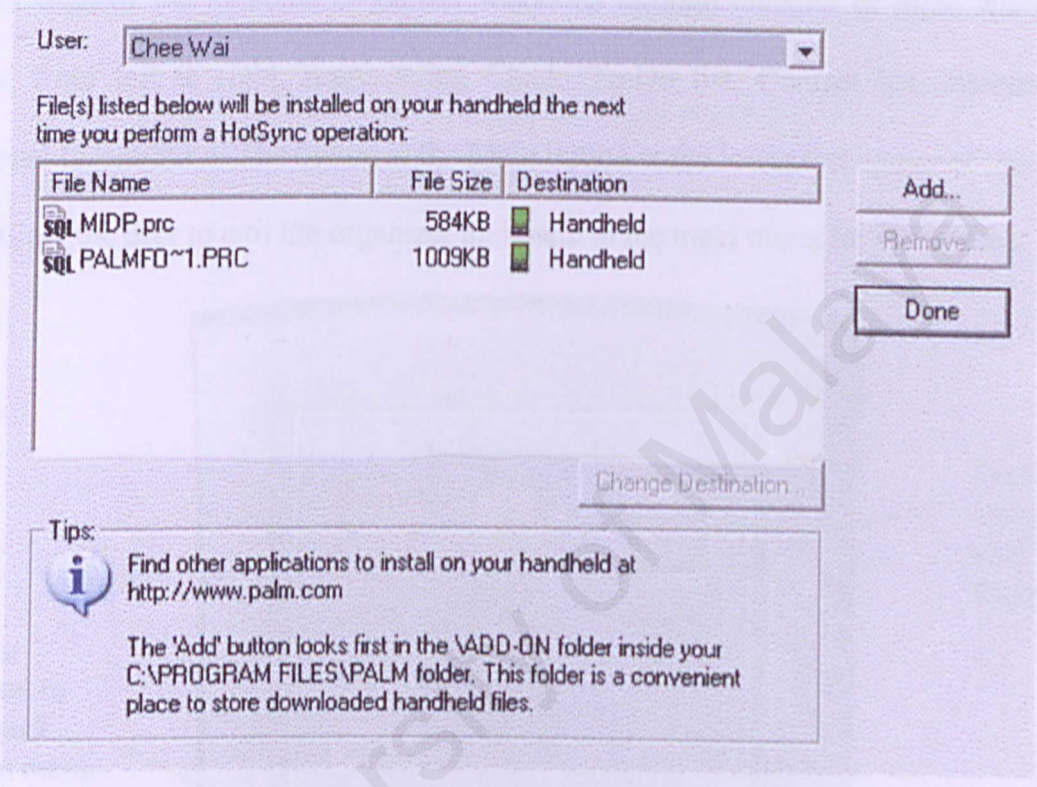


Figure 1.2: PalmForce loaded into the conduit and ready to be installed

After both of the prc file is loaded in the conduit, it can be installed into the device now by synchronizing it. After that, just click on the PalmForce icon in the Palm device's menu to start the application.

3.0 Personal Organizer Components

3.1 Main Menu

The Main Menu for the Personal Organizer consists of 5 icons and 1 button. User can access each of the Personal Organizer functions through clicking of these icons. The icons, from left to right, representing Clock, To-Do list, Contact list, Notepad and Personal Information. The button is the Main button at the lower left corner of the screen that allow the user to exit the organizer and back to the main menu for PalmForce.

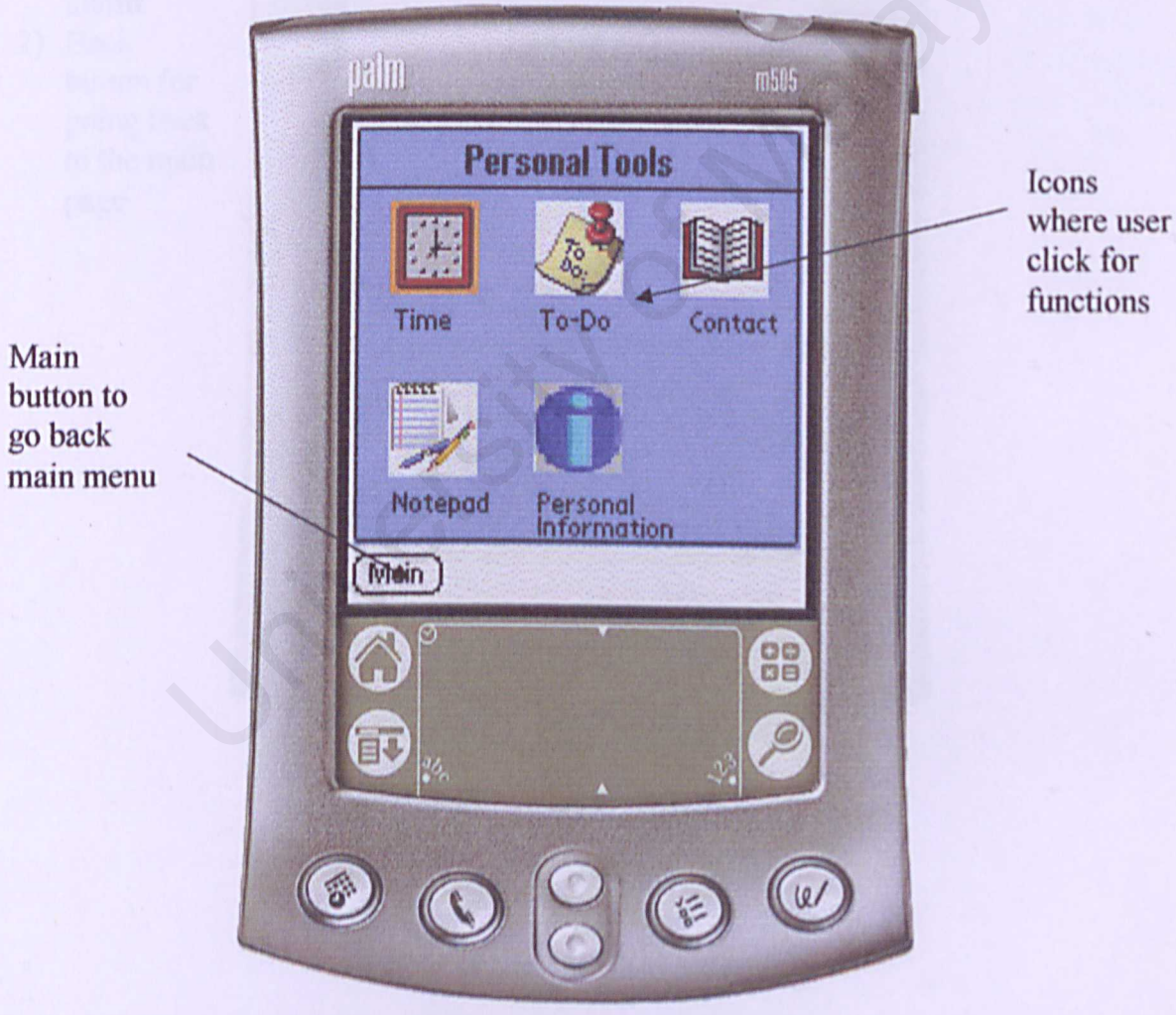


Figure 3.1: The main menu for Personal Organizer

3.2 Clock

Clock is the function where users use for time management. The Clock consist of a digital clock where user can get the date, time and day of the week and an alarm where user can set the time for the alarm. Below is the screen for Clock.

1) Alarm button for access the alarm.
2) Back button for going back to the main page.

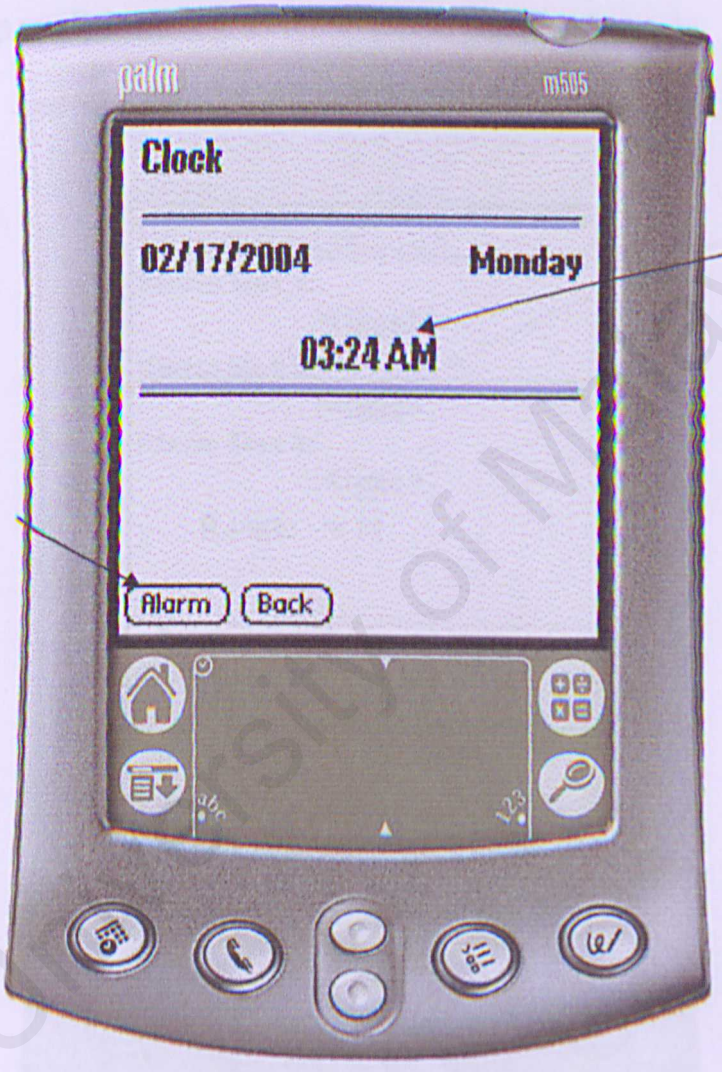


Figure 3.2: The Clock for Personal Organizer

3.3 Alarm

For the alarm function, there are 3 slots for users to set their time. The time slot will pop up to allow the user to insert the time. When the user wants to reset the time, they can just choose the slot they want to reset and click the reset button. When the time comes a alert box will pop up. Below is the screen to set the alarm.

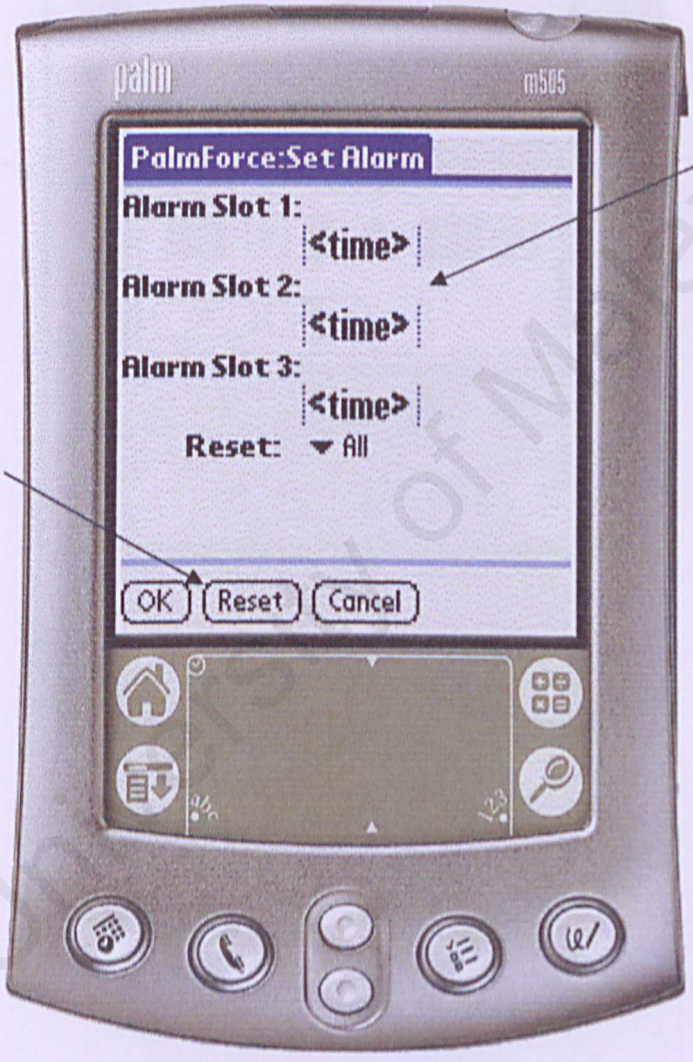
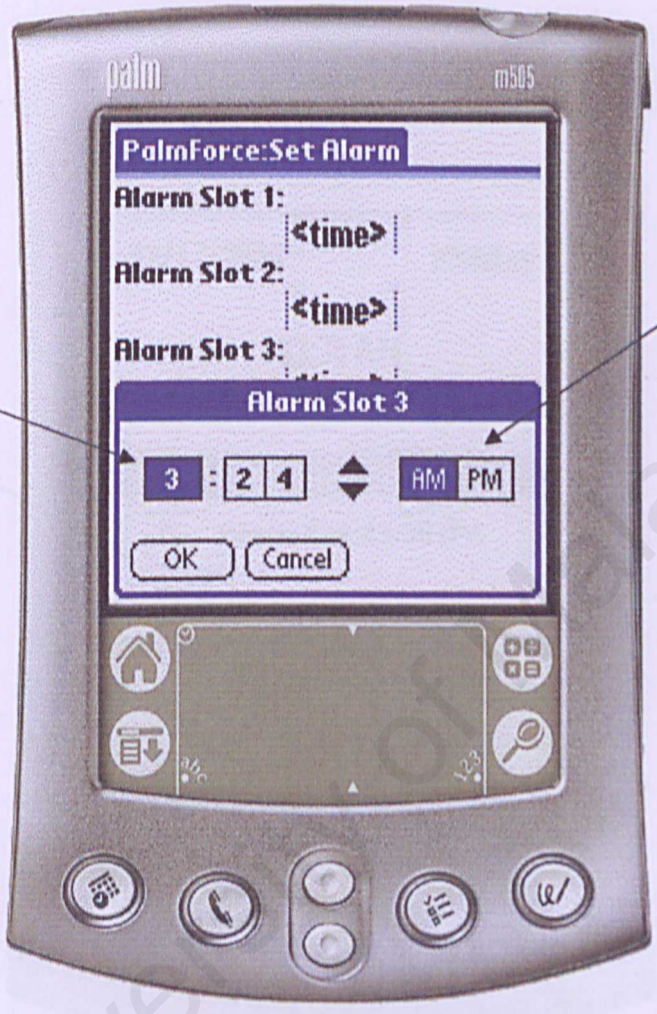


Figure 3.3: The Alarm form

3.3.1 Alarm : Setting the time

This is the time slot to select the time.

1) Set the time
2) OK button for select the time
3) Cancel button for exit the slot without selecting the time



Click to select the AM or PM section for the time

Figure 3.4: How to set the time

3.3.1 Alarm –The alarm box

This is how the alarm box will pop up.

Click done to close the alarm box

Figure 3.5: The alarm box

3.4 To-Do List

To-Do list is where users can write down their to-do and set the reminder for it. The table consist of 3 column displaying the date, time and the event. Below is the description for the main screen for To-Do list.

1) '>' for next page
2) '<' for previous page
3) 'New' for enter new event
4) 'Home' for going back to main screen

1) Display the Date, Time and To-Do event
2) Click on the row for details of the event

Figure 3.6: The To-Do list

3.4.1 To-Do List - Edit To-Do

This is the page to enter the To-Dos. The users could check the checkbox if they want to set the reminder for this event.

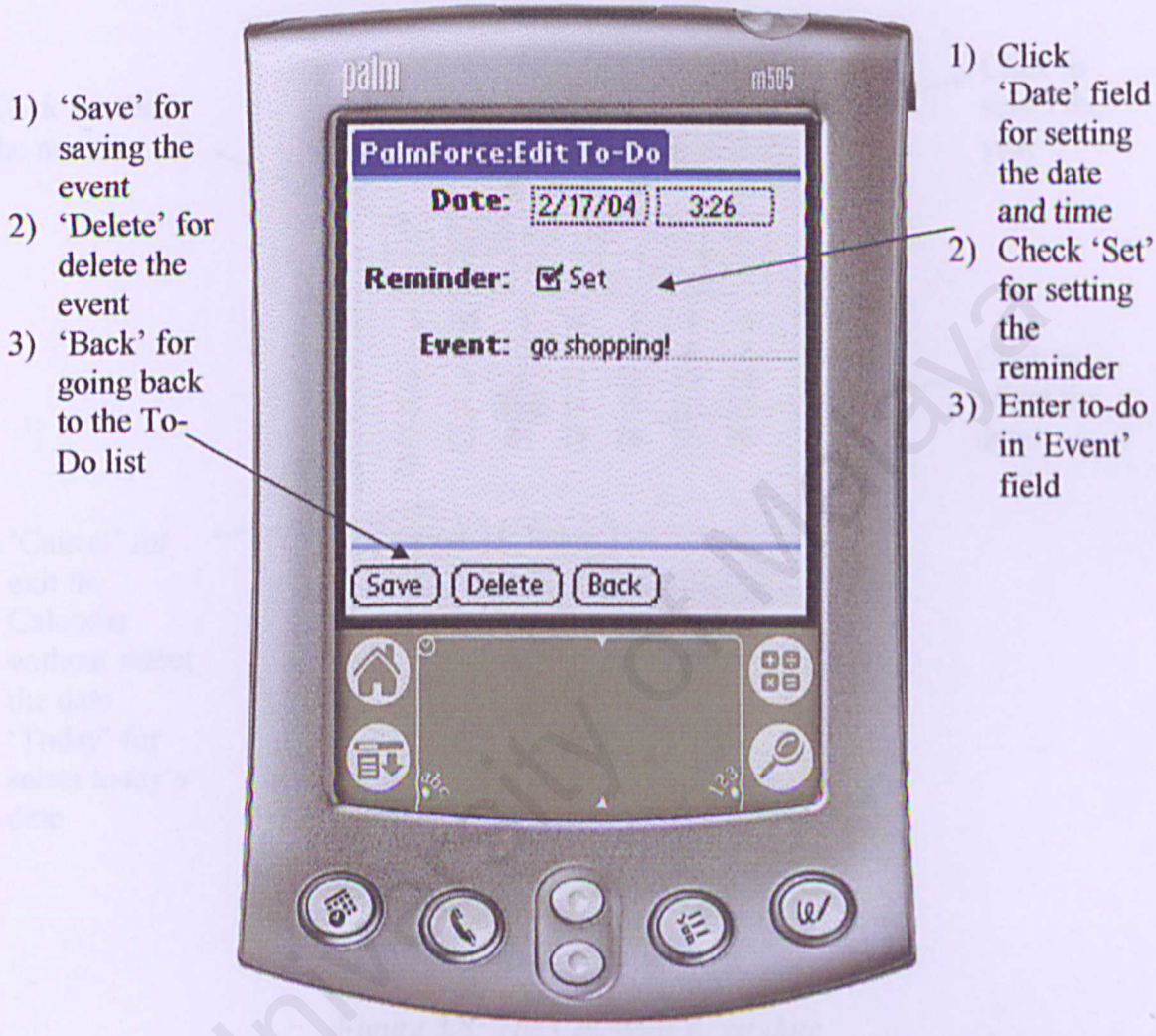


Figure 3.7: Edit To-Do

3.4.2 To-Do List – Setting the date

This is how the date is chosen in the Calendar. It would apply to ever Date set function in this module.

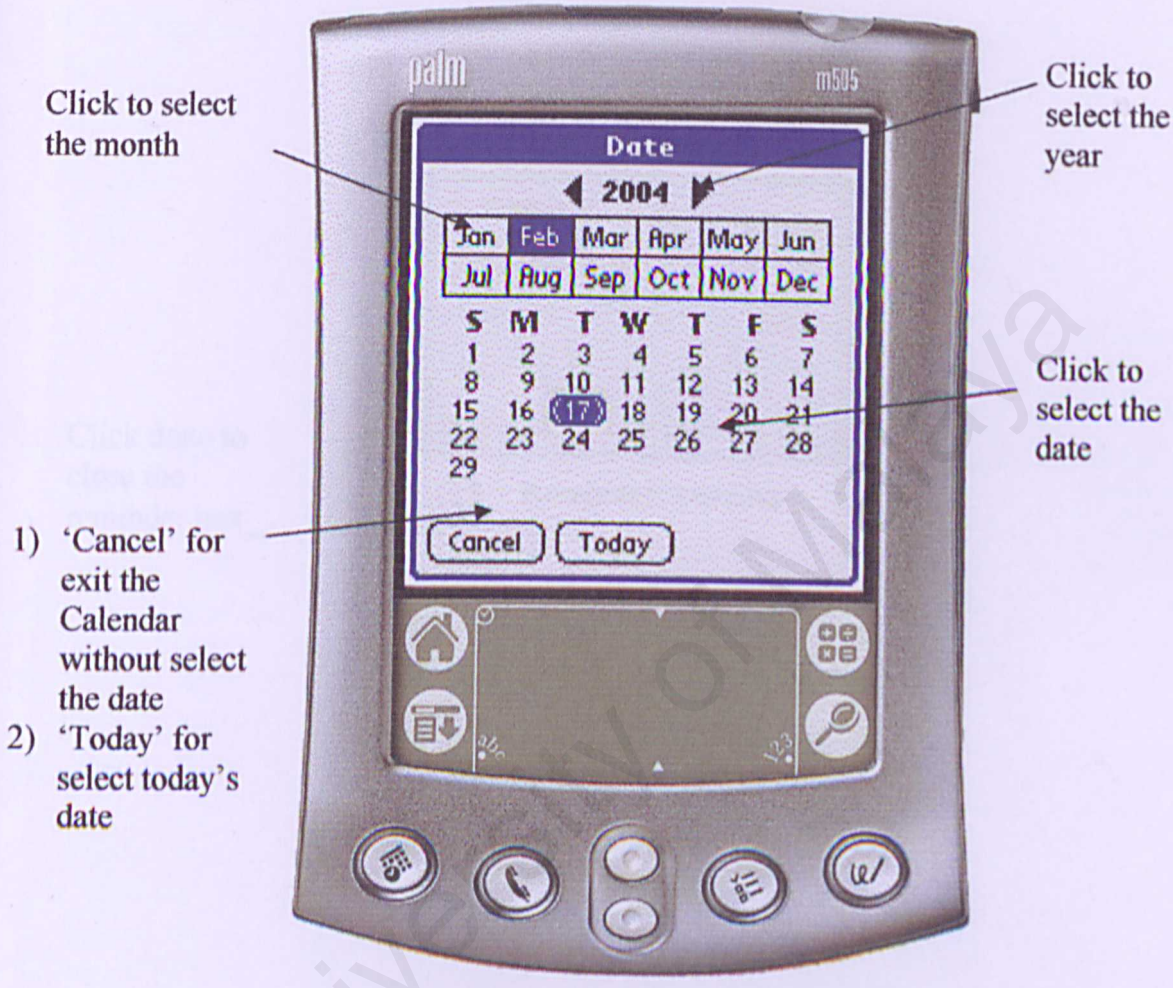


Figure 3.8: The Calendar to set date

3.4.3 To-Do List – The reminder box

This is how the reminder box would look like.

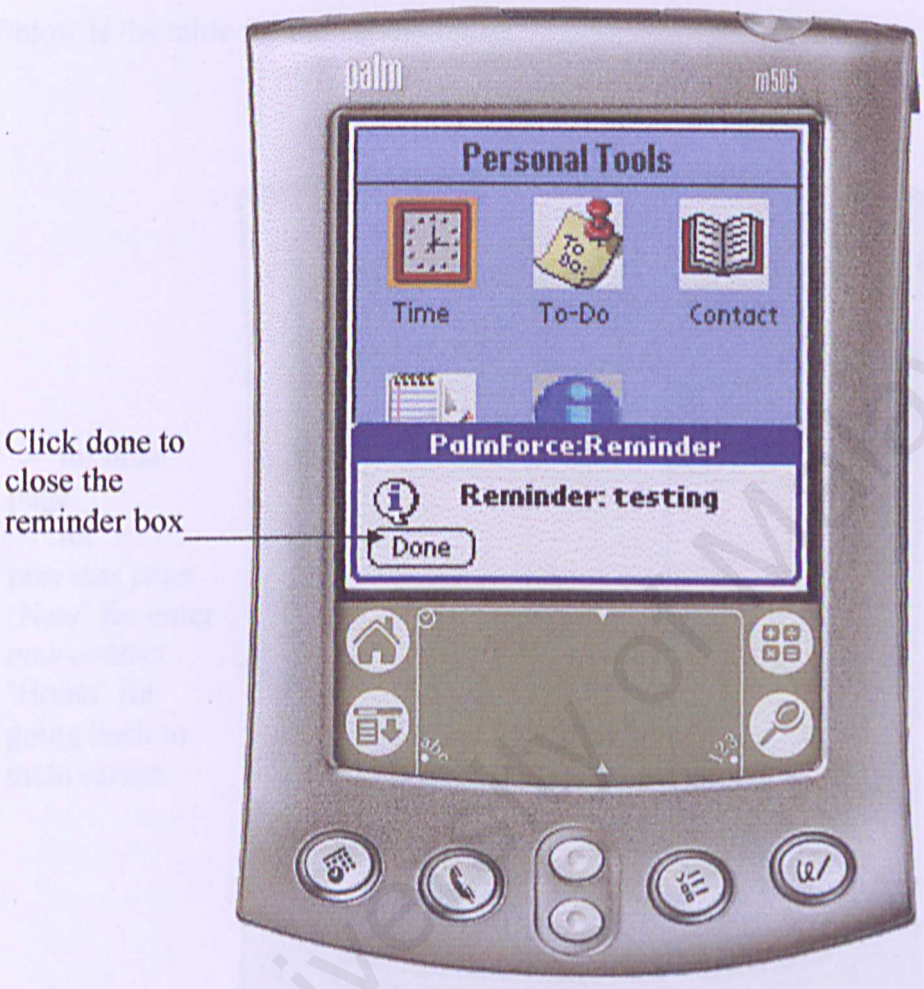
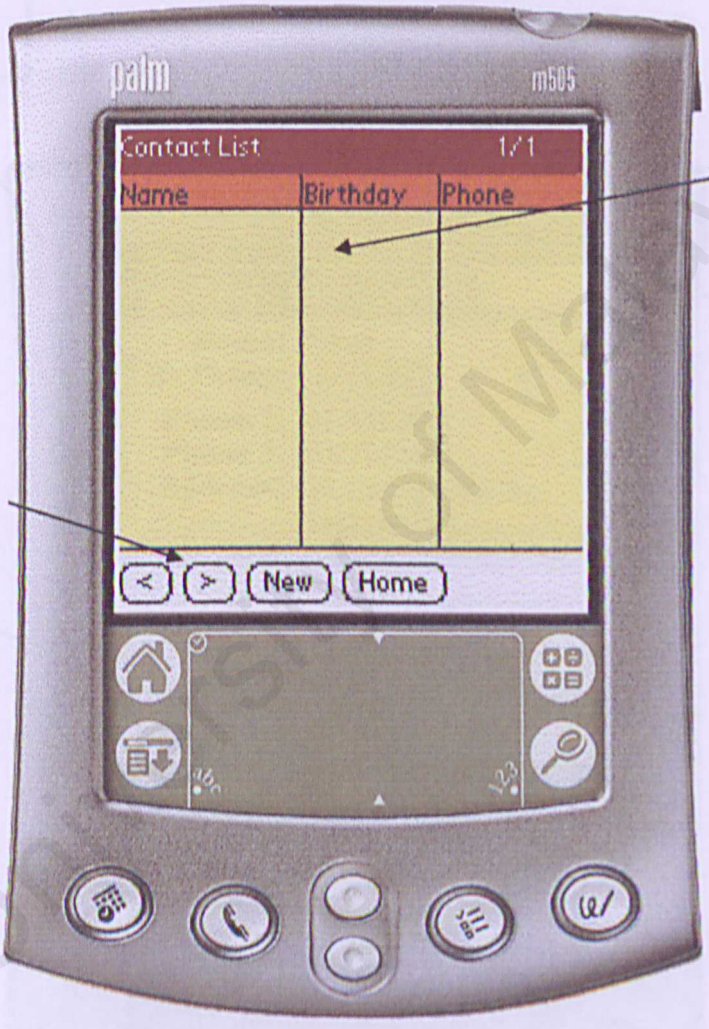


Figure 3.9: The reminder box

3.5 Contact List

The Contact list is similar to the reminder list. It allow users to edit and view their contact details. There would be a form to allow user to enter the detail of their contact. Below is the table for the list of contact.



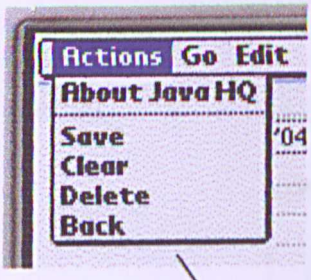
1) '>' for next page
2) '<' for previous page
3) 'New' for enter new contact
4) 'Home' for going back to main screen

1) Display the Name, Birthday and Phone of contact
2) Click on the row for details of the contact

Figure 3.10: The Contact list

3.5.1 Contact List – Edit Contact

This is form where users can edit their contact details. Note that the back button is placed on the menu on top of the palm.



This is where the 'Back' command located

Enter the Name, Birthday, Phone and Address of the contact

1) 'Save' to save the details
2) 'Clear' to clear the detail except the name
3) 'Delete' to delete the record

Figure 3.11: Edit To-Do

3.6 Notepad

This is where user can write their notes and save them accordingly. Users then can open the file by choosing the file they want. Below is the text box where users can write their notes. Note that the exit button is on the top menu bar, as mentioned in the Contact List section.

1) 'New' to open a new blank notes
2) 'Open' to open file
3) 'Save' to save the file

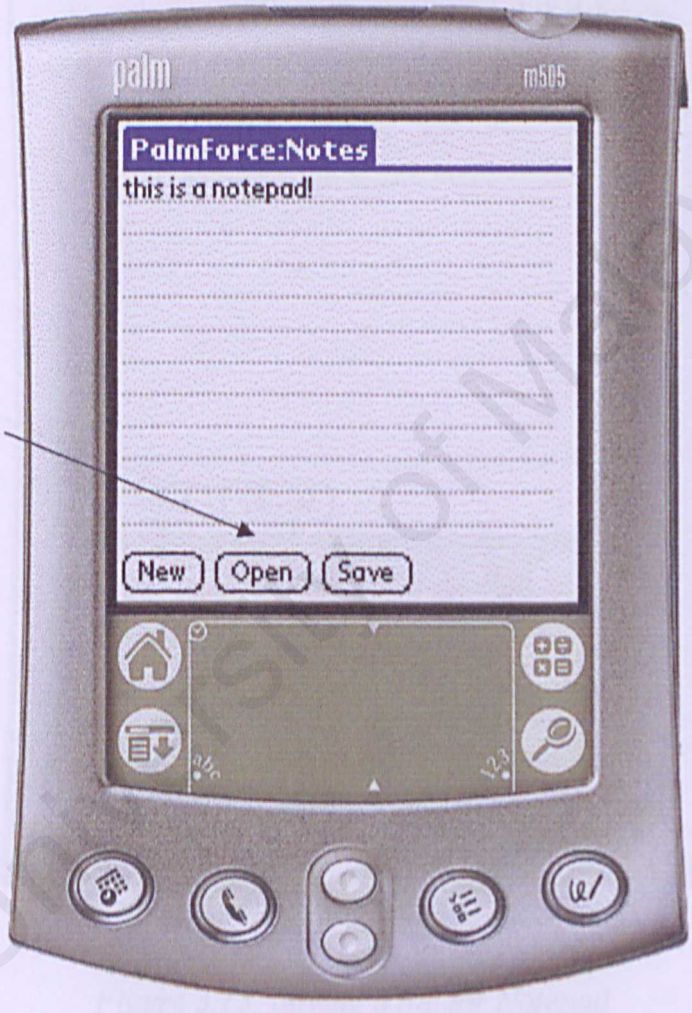


Figure 3.12: The Notepad

3.6.1 Notepad – Saving a file

This is where users save their file. They are required to enter a file name first.

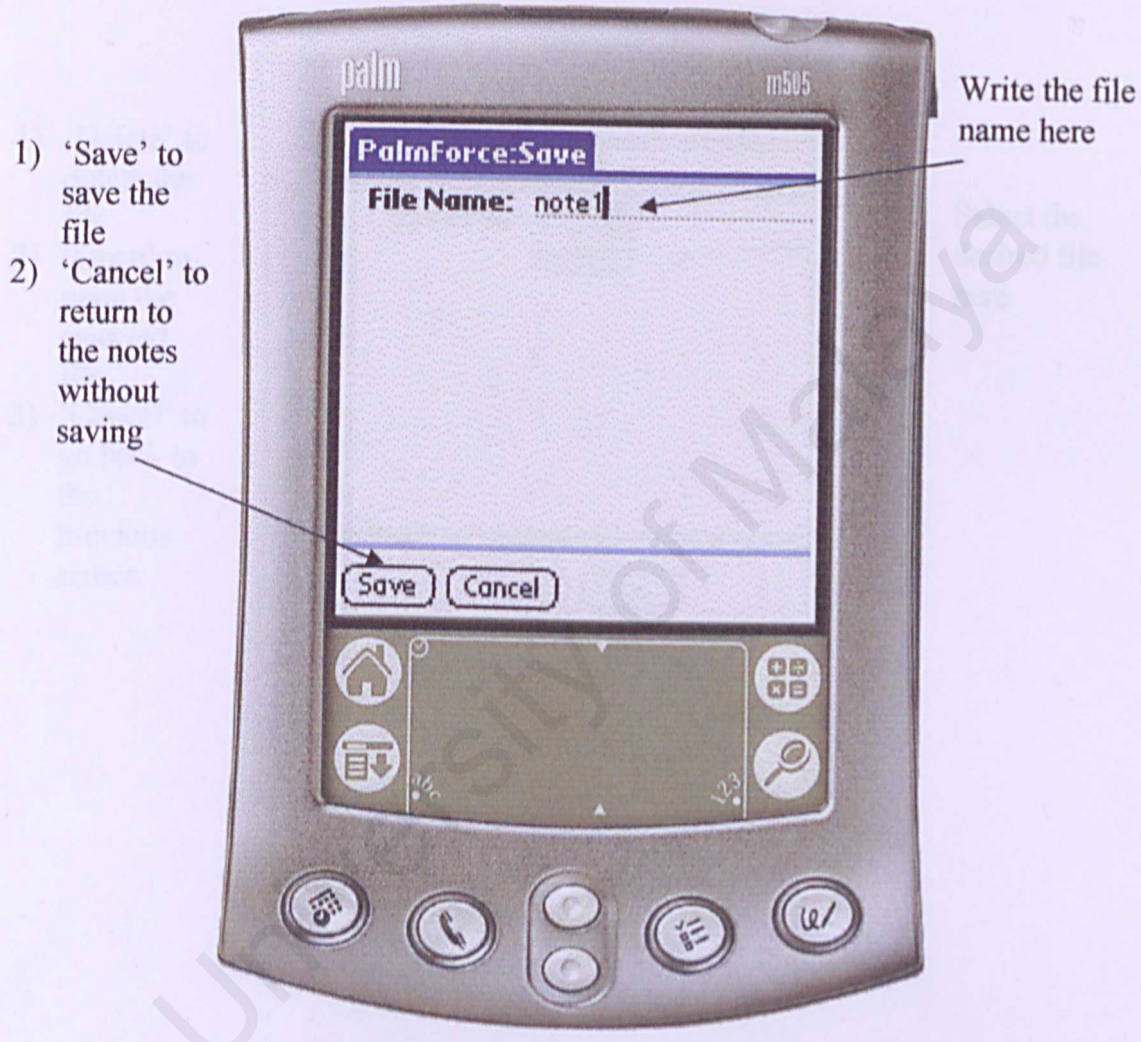
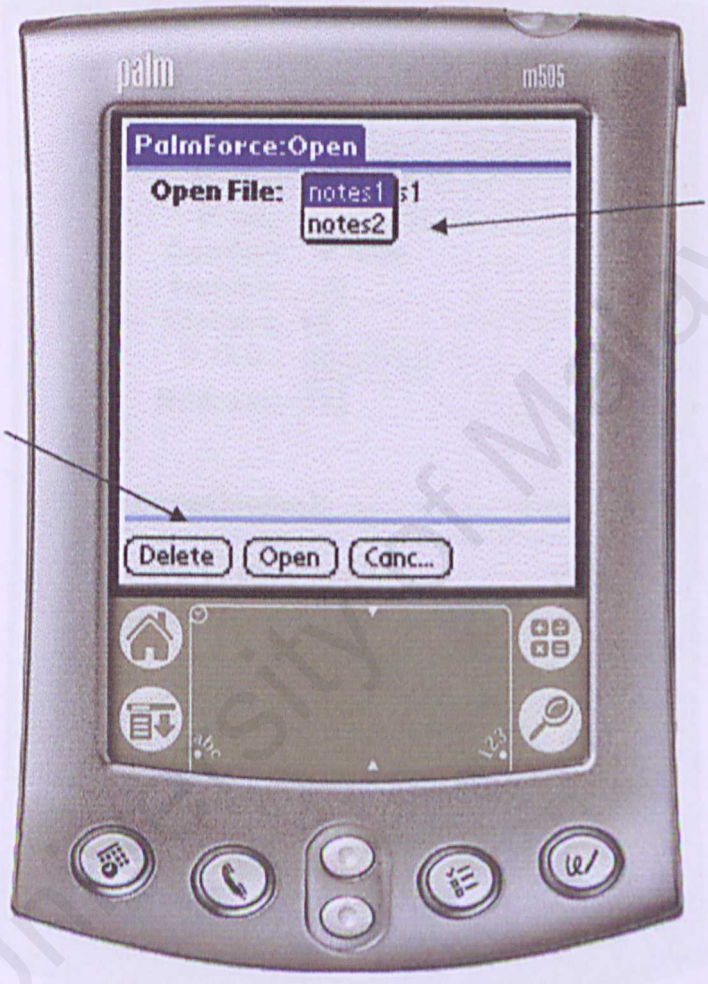


Figure 3.13: Saving a file for Notepad

3.6.2 Notepad – Opening a file

This is where users can open a file saved previously. The drop down menu contains the file name that is saved in the device. Users can select one of them and click 'Open' to open it or 'Delete' to delete the file.

1) 'Delete' to delete the file
2) 'Open' to open the selected file
3) 'Cancel' to go back to the previous screen



Select the desired file here

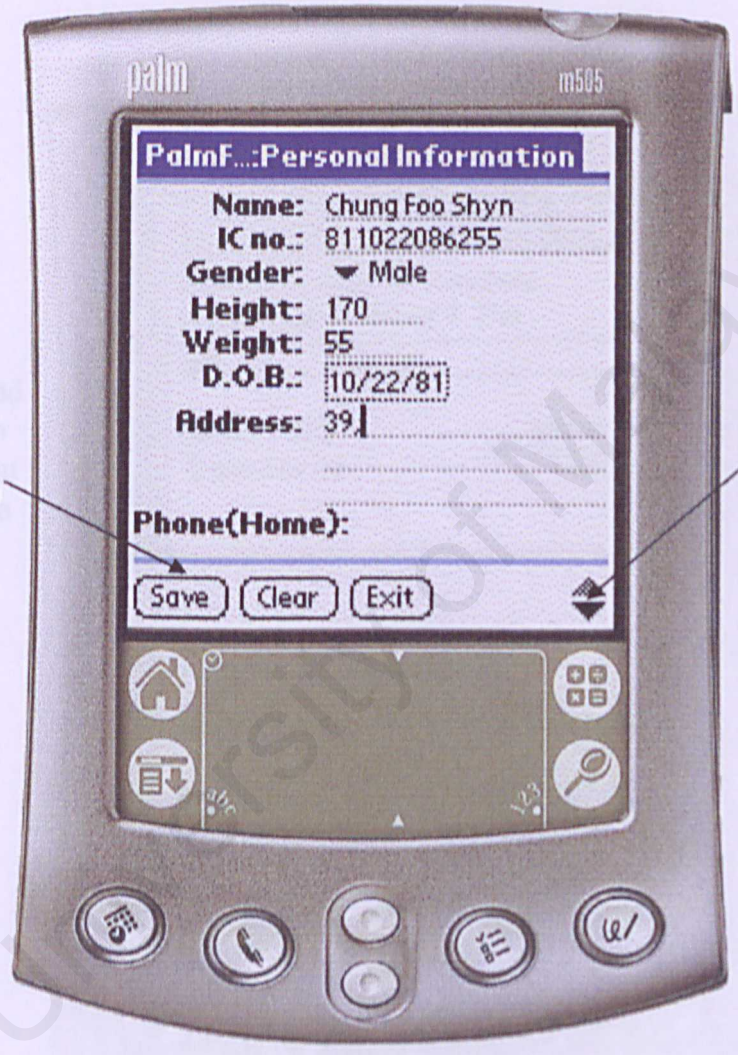
Figure 3.14: Opening a file for Notepad

3.7 Personal Information

This is where the users can enter their personal information for reference in the future.

Some fields such as Height and IC number are set to accept numeric inputs only.

1) 'Save' to save the details
2) 'Clear' to clear the details
3) 'Exit' to go back main menu



Click the arrow to navigate to different fields

Figure 3.15: Personal Information

The details that can be save includes: Name, IC no, Gender, Height, Weight, Date of Birth, Address, Home Phone, Mobile Phone and Passport no.

4.0 Uninstall PalmForce

To uninstall PalmForce on the device, just simply go to the delete page of PalmOS and select the PalmForce application. Then users can click on the delete button to uninstall it.

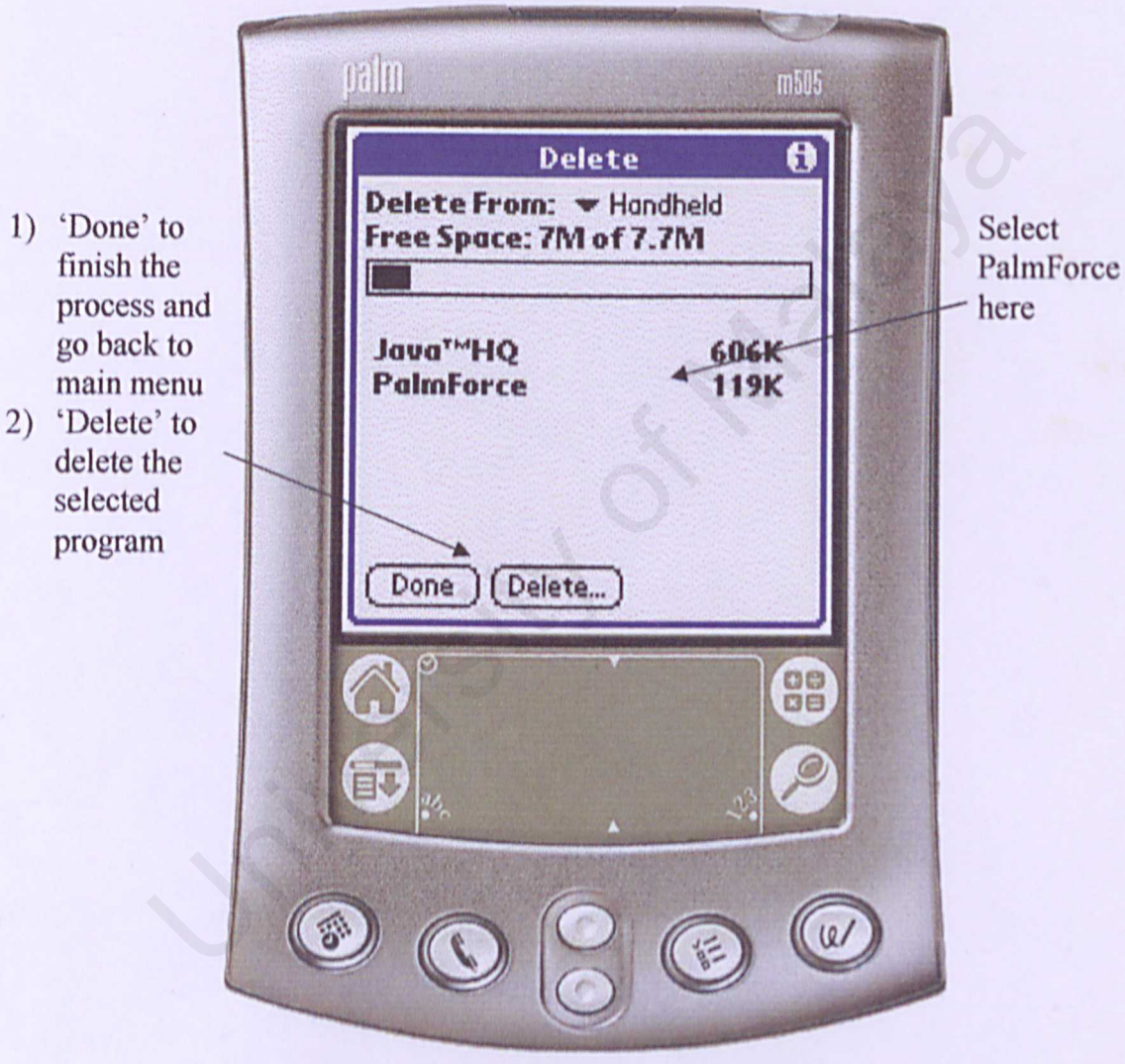


Figure 4.1: Uninstall PalmForce